VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

SYNCHRONIZACE TEXTU A AUDIA

TEXT TO AUDIO ALIGNMENT

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **ADAM ŠUBA**
AUTHOR

VEDOUCÍ PRÁCE **Ing. IGOR SZŐKE, Ph.D.**
SUPERVISOR

BRNO 2018

Zadání bakalářské práce/20916/2017/xsubaa00

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií Akademický rok 2017/2018

Zadání bakalářské práce

Řešitel: **Šuba Adam**
Obor: Informační technologie
Téma: **Synchronizace textu a audia**
Text to Audio Alignment
Kategorie: Zpracování řeči a přirozeného jazyka

Pokyny:

1. Nastudujte principy automatického zarovnání textu k audiu.
2. Navrhněte postupy, jak automaticky zarovnat text s odpovídajícím audiem. Uvažujte i postupy nezávislé na jazyku.
3. Implementujte navržené postupy. Zarovnejte vybraná data a porovnejte úspěšnost jednotlivých postupů.
4. Identifikujte místa (příčiny) selhání navržených postupů a pokuste se je vylepšit.
5. Znodnoťte dosažené výsledky a navrhněte směry dalšího vývoje.
6. Vyrobte A2 plakátek a cca 30 vteřinové video prezentující výsledky vaší práce.

Literatura:

- Dle pokynů vedoucího

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 a 2, a část bodů 3 a 4 ze zadání.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Szőke Igor, Ing., Ph.D.**, UPGM FIT VUT
Datum zadání: 1. listopadu 2017
Datum odevzdání: 16. května 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
612 66 Brno, Božetěchova 2

L.S.

doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Tato bakalářská práce se zabývá výzkumem nástroje pro synchronizaci textu a audia na úrovni jednotlivých grafémů a fonémů. V práci jsou také diskutovány možné přístupy k synchronizaci a případná omezení a problémy, kterým je třeba čelit. Zkoumaný nástroj využívá přístup vycházející z grapheme-to-phoneme konverze s použitím joint-sequence modelů. Pro experimenty jsou použity data z televizního vysílání, která byla převzata z Multi-Genre Broadcast Challenge 2015.

Abstract

This bachelor thesis studies a tool for automatic text to audio alignment at the level of single phonemes and graphemes. It also discusses possible techniques used in alignment and possible limitations and difficulties that need to be taken into account. Studied tool uses approach based on graphme-to-phoneme conversion using joint-sequence models. Data used in experiments are TV broadcast recordings from Multi-Genre Broadcast Challenge 2015.

Klíčová slova

synchronizace textu a audia, zarovnání, fonémový rozpoznávač, grapheme-to-phoneme konverze, g2p, MGB Challenge

Keywords

synchronization of text and audio, alignment, phoneme recognition, grapheme-to-phoneme conversion, g2p, MGB Challenge

Citace

ŠUBA, Adam. *Synchronizace textu a audia.* Brno, 2018. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Igor Szőke, Ph.D.

Synchronizace textu a audia

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Igora Szőkeho, Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Adam Šuba
16. května 2018

Poděkování

Děkuji panu Ing. Igorovi Szőkemu, Ph.D. za odbornou pomoc při řešení této práce, dodaná data a všechny další rady.

Obsah

Kapitola 1

Úvod

Tato práce se zabývá automatickou synchronizací textu a audia, tedy přiřazením časových značek k jednotlivým slovům v textovém přepisu tak, aby tyto časy odpovídaly výskytu těchto slov ve zvukové nahrávce. Úloha nachází uplatnění při indexaci multimediálních dat za účelem vyhledávání [13], při automatickém časování manuálně vytvořených titulků k filmům či seriálům nebo při synchronizaci audio knihy k její elektronické textové verzi [3].

V rámci této práce je zkoumán nástroj pro zarovnávání textu a audia na úrovni jednotlivých grafémů a fonémů. Účelem zkoumání je odhalení slabých míst tohoto přístupu a této konkrétní implementace pro úlohu synchronizace textu a audia. Přístup vychází z úlohy grapheme-to-phoneme konverze, zabývající se hledáním výslovnosti textu, která je nalezena právě jako posloupnost fonémů odpovídající dané posloupnosti grafémů.

Principy, které lze v úloze synchronizace uplatnit, a přehled prací, které se synchronizací zabývali, se nachází v kapitole 2. Kapitola 3 pojednává o Multi-Genre Broadcast Challenge, ze které tato práce čerpá testovací data a způsob skórování. Obecný postup zarovnávání, adaptování a trénování modelů a seznam použitých nástrojů se nachází v kapitole 4. Průběh provedených experimentů s jejich výsledky je obsahem kapitoly 5. Závěrečná kapitola 6 provádí shrnutí a zhodnocení dosavadní práce a diskutuje další možný výzkum.

Kapitola 2

Principy synchronizace textu a audia

Problematikou synchronizace textu a audia, neboli (automatického) zarovnání textu a audia, se již v minulosti zabývala celá řada prací. Náročnost této úlohy a techniky, které lze pro řešení použít, mohou být ovlivněny následujícími skutečnostmi:

- délka nahrávek – se zvyšující se délkou nahrávek (a tedy i textu) rostou jak paměťové, tak časové nároky, pro delší nahrávky je nutné volit úspornější algoritmy
- kvalita nahrávek – nízká akustická kvalita nahrávek může způsobovat problémy při použití metod spoléhajících se na přepis řeči na text a následné zarovnávání na textové úrovni
- kvalita a přesnost textu – pokud zarovnávaný text obsahuje chyby, neobsahuje všechna vyslovená slova nebo obsahuje jinak formulované věty (jev častý například v titulcích), není možné text přesně zarovnat
- vyžadovaná přesnost zarovnání – zarovnání může probíhat na různých úrovních: odstavce, věty, slova nebo samotná písmena, může být také tolerována chyba zarovnání v řádech milisekund až sekund
- hudba na pozadí – pokud nahrávky obsahují kromě mluveného slova i hudbu obsahující zpěv, který často nebývá součástí zarovnávaných přepisů, může to způsobovat problémy
- závislost na jazyku – spousta postupů je závislá na znalosti jazyka, který je zarovnáván
- ticho – v nahrávkách se mohou vyskytovat dlouhá tichá místa, která nejsou v textovém přepisu nijak vyznačena

Jako příklad lze uvést systém pro automatické zarovnání elektronické a audio knihy. Takový systém například nevyžaduje příliš velkou odolnost proti chybějícím nebo rozdílným slovům v textovém přepisu z toho důvodu, že audio kniha ze své podstaty odpovídá přesně své textové verzi (elektronické knize). Navíc bývají audio knihy nahrávány herci v profesionálních studiích a výsledná kvalita nahrávek je na vysoké úrovni. Oproti tomu, systém pro zarovnávání titulků k filmům nebo seriálům se musí vypořádat s texty, kde jsou některá slova vynechána a věty mohou být jinak formulovány (z důvodu rychlých promluv, případně zjednodušené titulky). Navíc se v nahrávkách často bude vyskytovat hudba a různé jiné hluky.

2.1 Grafém a foném

Jak již název práce napovídá, používaná technika využívá zarovnání na úrovni fonémů a grafémů (oproti zarovnání na úrovni vět či slov). Foném je nejmenší jednotka řeči, která rozlišuje v daném jazyce jedno slovo od druhého. Grafém je naopak nejmenší nedělitelná jednotka psané formy daného jazyka. Grafémem mohou být jednak písmena, ale také číslice, čínské znaky nebo interpunkční znaménka. Foném může být reprezentován jedním nebo více grafémy. Například české písmeno *ch* je složeno ze dvou grafémů *c* a *h* a jedná se o jeden foném, který je v mezinárodní fonetické abecedě IPA (Internation Phonetic Alphabet) značen jako `[x]`.

2.2 Přehled prací zabývajících se synchronizací

Tato sekce předkládá stručný přehled některých dřívějších prací zabývajících se problematikou automatického zarovnání audia a jemu odpovídajícího textu. Byly vybrány jen stěžejní práce a ty, které blíže souvisejí s tématem této práce.

2.2.1 Automatic Generation of Hyperlinks between Audio and Transcript

J. Robert-Ribes a R.G. Mukhtar [13] navrhli v roce 1997 prototyp systému pro automatické propojování textových přepisů a audia pomocí odkazů v textu. Systém měl být použit v systému FRANK (Filme Researchers Archival Navigation Kit), sloužícímu k prohledávání archívů digitálních videí. Navrhovaný systém přidával značky na úrovni odstavců s tolerovanou chybou až 2 sekundy, pracoval off-line bez omezení na délku nahrávek, dokázal pracovat s hudbou a šumem na pozadí a nepřesným textovým přepisem.

Navržený systém pracuje následovně: Nejprve rozdělí přepis i audio na segmenty, ze segmentů přepisu vygeneruje jazykový model, který se použije pro automatické rozpoznávání řeči na segmentu audia. Výstup rozpoznávače se v posledním kroku pokusí zarovnat na textový segment. Systém iterativně upravuje okno pro segmentaci podle úspěchů či neúspěchů zarovnání.

Při testování nebyl systém schopen vytvořit zarovnání pro 8,7 % odstavců, ve zbytku nebyla chyba zarovnání větší než 3 sekundy.

2.2.2 A Recursive Algorithm for the Forced Alignment of Very Long Audio Segments

V roce 1998 reaguje Pedro J. Moreno a další [11] na problém využití Viterbiho algoritmu pro zarovnání dlouhých nahrávek, kdy razantně rostou paměťové a časové nároky. Autoři navrhují rekurzivní algoritmus, který využívá jazykového modelu vytvořeného ze zarovnávaného přepisu k rozpoznání řeči a vytvoření automatického textového přepisu. Ten je technikami dynamického programování zarovnán s manuálním přepisem. Cílem není nalézt perfektní zarovnání celé nahrávky, ale najít pouze tzv. ostrovy důvěry (islands of confidence), které slouží k rozdělení nahrávky do menších segmentů. Na takto získaných segmentech pokračuje algoritmus rekurzivně ve svojí činnosti.

Algoritmus byl použit k indexování multimediálních dat. Experimenty ukázaly, že 98,5 % slov bylo zarovnáno s chybou do 0,5 sekundy a 99,75 % slov s chybou do 2 sekund.

2.2.3 Alignment of Speech to Highly Imperfect Text Transcriptions

Pánové Habould a Kender se ve své práci [8] zaměřují na zarovnání audia ze záznamů přednášek, které obsahuje několik různých mluvčí s různými dialekty, k automaticky vytvořeným přepisům. Přepisy mají word error rate (WER) až 70 %.

Ze zvukové stopy je vytvořena množina vyskytujících se fonémů. Z přepisu jsou za pomocí výslovnostního slovníku také získány fonémy. Zarovnání probíhá globálně, podobným způsobem jaký se používá při zarovnávání sekvencí DNA. Fonémů z textových přepisů je podstatě méně a jsou k fonémům z audia zarovnány s pomocí kopírování, přidávání, mazání a zaměňování, ovšem bez přemisťování. K ohodnocení kvality zarovnání se využívá edit distance, která hodnotí podobnost dvou řetězců [10]. Po zarovnání fonémů na sebe jsou tyto informace přeneseny zpět do původního textového přepisu.

Při experimentech se podařilo správně zarovnat 60 % slov s chybou do 10 s. Vzhledem ke kvalitě automaticky vytvořených přepisů se nejedná o špatné výsledky.

2.2.4 Automatic Synchronization of Electronic and Audio Books via TTS Alignment and Silence Filtering

Cílem práce [3] z roku 2011 bylo vytvořit aplikaci pro iOS pro čtení elektronických knih s možností poslechu audio verze během čtení. Jak již bylo v úvodu této kapitoly nastíněno, synchronizace elektronických a audio knih je úloha poněkud snadnější.

Navržený algoritmus implementovaný ve výsledné aplikaci je následující: Textový přepis je převeden na řeč (text-to-speech, TTS). Z obou zvukových stop jsou filtrováním odstraněna tichá místa. Obě takto připravené nahrávky jsou na závěr zarovnány pomocí dynamického borcení času (DTW).

2.2.5 Text-to-Speech Alignment of Long Recordings Using Universal Phone Models

Sarah Hoffmann a Beat Pfister se ve své práci [9] zaměřují na jazykově nezávislé zarovnání audia a textu. Využívá obecnou sadu skrytých Markovových modelů (HMM), které jsou použity jako obecný model k zarovnání audia a textu na úrovni vět. Jakmile je získána segmentace na věty, může být použito dalších postupů pro další segmentaci.

Postup je následující: V textovém přepisu jsou označena potencionální místa, která oddělují jednotlivé věty. Text je poté převeden do fonémové podoby s tichými místy, reprezentujícími získané okraje vět. Na závěr je řečový signál zarovnán s fonémovým přepisem.

HMM byly natrénovány na třech jazycích: němčina, angličtina a francouzština. K testování byly využity přepisy s velmi nízkou chybovostí (WER, 0.61 %). Byly zarovnávány jazyky němčina, angličtina, francouzština, finština a bulharština. Pro každý přepis byl vygenerován grapheme-to-phoneme model (viz 2.3.1) pomocí dat z anglické verze Wiktionary slovníku[1]. Tyto modely byly použity pro konverzi textových přepisů na sekvence fonémů. Okraje vět byly určeny podle výskytu teček, otazníků a vykřičníků.

Výsledné zarovnání bylo na dobré úrovni i pro jazyky, na kterých nebyly HMM natrénovány. Autoři diskutují použití sofistikovanějších postupů pro detekci okrajů vět, pro zlepšení výsledků.

[1] https://en.wiktionary.org/

2.2.6 Shrnutí

Jak je vidět z předchozích sekcí, přístupů k synchronizaci textu a audia je mnoho a jejich výčet definitivně není kompletní. Velmi často bývá používaný Viterbiho algoritmus, který není efektivní pro velmi dlouhé nahrávky. Tento problém lze řešit například rekurzí. Většina prací zarovnává na textové úrovni, což vyžaduje nástroje pro rozpoznávání řeči. U těchto přístupů bývají často dále používány skryté Markovovy modely nebo vážené stavové převodníky. Další možností se zdá být zarovnávání na úrovni audia, kdy je pomocí TTS nástroje textový přepis převeden na řeč a ta je k původnímu audiu zarovnána dynamickým borcením času. Různé metody vyžadují specifické podmínky na zarovnávaná data, tak jak již bylo nastíněno v úvodu této kapitoly.

Některé práce využívají, stejně jako technika používaná zde, fonémy. Nicméně například v případě [8] se jedná o zarovnání fonémů na fonémy, zatímco v této práci jsou fonémy zarovnávány na grafémy. Více o tomto přístupu se nachází v následujících sekcích.

2.3 Grapheme-to-phoneme alignment

Přístup, který používá tato práce se v angličtině nazývá Grapheme-to-phoneme alignment, česky lze tento pojem přeložit jako zarovnání grafému na foném. Jelikož není český název ustálen, budeme v dalším textu používat zkratku G2P.

G2P zarovnání znamená, že se snažíme nalézt nejpravděpodobnější namapování jednotlivých grafémů na fonémy, takové, že tyto fonémy jsou pak výslovností jednotlivých grafémů. Hledání výslovnosti textového přepisu provádí takzvaná grapheme-to-phoneme konverze (G2P konverze). V případě zarovnání je rozdíl takový, že máme již existující fonémový přepis a chceme v něm nalézt odpovídající výslovnost.

Následující sekce 2.3.1 rozebírá problematiku G2P konverze. Pokud chceme provádět zarovnání fonémů na grafémy, musíme nejprve z nahrávek získat jejich fonémový přepis, této problematiky se dotýká sekce 2.3.2.

2.3.1 Grapheme-to-phoneme konverze

G2P zarovnání vychází z původně odlišené úlohy a to tzv. G2P konverze, která spočívá v nalezení výslovnosti (tedy fonémového přepisu) slova zadaného v psané formě [5]. Tato úloha je snadná v případě, že má jazyk perfektně fonetický pravopis, tzn. existuje bijektivní vztah (vztah 1-ku-1) mezi grafémy a fonémy daného jazyka. Takový pravopis je ovšem velmi vzácný, mezi jazyky s téměř perfektním fonetickým pravopisem patří srbochorvatština či esperanto.

Techniky Grapheme-to-phoneme konverze

Slovníkové vyhledávání (dictionary look-up) je nejjednodušší technikou G2P konverze, spočívá v jednoduchém vyhledání ve slovníku a vrácením odpovídající sekvence fonémů. Nevýhodou je zdlouhavá tvorba slovníku a jeho paměťová náročnost. Navíc slovník s konečným počtem záznamů má vždy limitované možnosti.

Pravidlové (rule-based) systémy využívají pravidel často ve formě konečných automatů. Mohou obsahovat také slovník jako seznam výjimek. Nevýhodou tohoto přístupu je náročnost na tvorbu pravidel a nutnost definovat výjimky z pravidel, které bývají v jazycích běžné.

Daty řízené (data-driven) systémy vychází z toho, že podobná slova budou mít analogicky podobou výslovnost. Stačí tedy systému předat sadu trénovacích dat, která obsahuje příklady odpovídající pravidlům a výjimkám. U těchto systémů je zásadní otázkou způsob implementace zmíněné analogie. Existují tři hlavní přístupy:

- lokální klasifikace – pro každý vstupní znak je zvolena sekvence fonémů z malé množiny povolených, výstup je predikován podle kontextu aktuálního znaku
- podobnost – na rozdíl od lokální klasifikace je bráno v potaz celé slovo a hledá se nejpodobnější slovo nebo část slova v trénovací sadě
- pravděpodobnostní přístupy

Joint-sequence modely

V této sekci je naznačena matematická podstata joint-sequence modelů, které se používají pro daty řízené G2P konverze. Popis vychází [5].

Pro množinu grafémů G a sadu fonémů Φ můžeme formálně definovat úlohu G2P konverze jako

$$\boldsymbol{\phi}(\boldsymbol{g}) = \underset{\boldsymbol{\phi}' \in \Phi^*}{\operatorname{argmax}}\, p(\boldsymbol{g}, \boldsymbol{\phi}')$$

Hledáme tedy pro danou písemnou formu $\boldsymbol{g} \in G^*$ nejpravděpodobnější výslovnost (sekvenci fonémů) $\boldsymbol{\phi} \in \Phi^*$. V^* je množina všech řetězců vytvořených nad množinou V.

Podstatou joint-sequence modelů je myšlenka, že vztah vstupních a výstupních sekvencí lze získat ze společné sekvence sdružených jednotek, kterým se říká *graphone* (spojení anglických slov *grapheme* a *phoneme*, česky můžeme nazvat jako grafón). Jedná se o páry $q = (\boldsymbol{q}, \boldsymbol{\phi} \in Q \subset G^* \times \Phi^*)$ sekvencí fonémů a grafémů o různých (i nulových) délkách. Ukázka rozdělení slova na jednu z možných sekvencí grafónů je na obrázku 2.1.

mixing	=	m	i	x	ing
[mɪksɪŋ]		[m]	[ɪ]	[ks]	[ɪŋ]

Obrázek 2.1: Ukázka rozdělení slova *mixing* na sekvenci grafónů

Tomu, že jsou grafémové a fonémové sekvence rozděleny do stejného počtu segmentů, se říká ko-segmentace. Pravděpodobnost $p(\boldsymbol{g}, \boldsymbol{\phi})$ je poté vyjádřena jako $p(\boldsymbol{g}, \boldsymbol{\phi}) = \sum_{\boldsymbol{q} \in S(\boldsymbol{g}, \boldsymbol{\phi})} p(\boldsymbol{q})$, kde $S(\boldsymbol{g}, \boldsymbol{\phi})$ je množina všech ko-segmentací $\boldsymbol{g}$ a $\boldsymbol{\phi}$.
Výpočet pravděpodobností je poté závislý od konkrétních modelů, např. [5] nebo [7]. Nástroj používaný v této práci vychází z [5].

2.3.2 Fonémový rozpoznávač

Aby bylo možné zarovnávat grafémy na fonémy je potřeba získat fonémový přepis zvukové nahrávky a i když není fonémové rozpoznávání předmětem této práce, v krátkosti shrneme princip, který se používá. Rozpoznávání fonémů z řečového signálu je samo o sobě široká disciplína zpracování řeči, která nachází uplatnění v mnoha úlohách. Můžeme z nich jmenovat například rozpoznávání řeči, vyhledávání klíčových slov, identifikaci jazyka či mluvčího a další. V této sekci si jen krátce nastíníme jak rozpoznávání probíhá, popis vychází z [14].

Proces rozpoznávání lze rozdělit do tří částí: *feature extraction*, *acoustic matching* a *decoder*. Vstupem části *feature extraction* je řečový signál, který je rozdělen do překrývajících

se rámců, obvykle o velikosti 25 ms a s posunem 10 ms. Z těchto segmentů jsou získány rysy popisující daný segment. Nejčastěji se dnes používá 13 mel-frekvenčních kepstrálních koeficientů (MFCC), získaných mel-frekvenční kepstrální analýzou.

V části *acoustic matching* dochází k přiřazení segmentů signálu k modelům jednotlivých fonémů vždy s určitou pravděpodobností. Tato část bývá implementována například pomocí skrytých Markovových modelů (HMM) v kombinaci buď s GMM (Gaussian Mixture Model) nebo s neuronovými sítěmi.

Během poslední části dochází k výběru nejlepší cesty přes jednotlivé fonémy a tedy k vytvoření řetězce fonémů reprezentujícího vstupní signál. Využívá se Viterbiho algoritmus pro hledání nejlepší cesty ve skrytém Markovově modelu.

Na výstupu fonémového rozpoznávače je sekvence fonémů v následujícím formátu:

```
0.00 0.06 d -25.430321
0.06 0.13 ah -12.715534
0.13 0.18 n -7.172920
0.18 0.21 hh -7.228844
0.21 0.27 ih -8.065895
0.27 0.32 z -7.863293
0.32 0.38 hh -12.804153
0.38 0.46 l -11.667976
0.46 0.64 ay -25.765778
0.64 0.72 hh -8.971939
```

Každý foném je na vlastním řádku, který také obsahuje čas začátku a konce daného fonému v sekundách a číslo udávající věrohodnost fonému.

Kapitola 3

Multi-Genre Broadcast Challenge

Data použitá v experimentech v této práci pochází z prvního ročníku tzv. Multi-Genre Broadcast (MGB) Challenge konaného v roce 2015. Z tohoto důvodu, se v této sekci s MGB Challenge seznámíme.

MGB Challenge je od roku 2015 každoročně konanou výzvou v oblastech rozpoznávání řeči, rozpoznávání řečníka, detekci dialektů a zarovnání textu k audiu. Jednotlivé úlohy se každý rok drobně liší. Výzva bývá konána v rámci workshopů zabývajících se technologiemi zpracování řeči. V roce 2015 to byl IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), v roce 2016 IEEE Spoken Language Technology Workshop (SLT) a v roce 2017 opět ASRU. Zmíněné workshopy jsou konány každé dva roky.

Výzva získala své jméno podle dat, které jsou pro účely jednotlivých úloh použity. Jedná se o široký výběr různých žánrů televizního vysílání [4]. Mezi specifika takových nahrávek patří jejich rozmanitost ve smyslu délky a kvality hlasové stopy (studiové nahrávky mají čistší zvuk než například reportáž ze sportovní události s jásajícími fanoušky v pozadí), výskyt hudby mezi dialogy nebo široká škála dialektů od různých řečníků.

I když data použité v této práci pochází z MGB Challenge 2015, podíváme se pro úplnost v následujících sekcích na souhrn informací o všech třech dosavadních ročnících.

3.1 MGB-1

V MGB-1, původně jen MGB Challenge, se bylo možné účastnit čtyř různých úloh:

1. Automatický přepis řeči na text (Text-to-speech transcription)
2. Zarovnání (Alignment) audia k titulkům
3. Dlouhodobý přepis řeči na text (Longitudinal speech-to-text transcription) s použitím několika epizod jednoho seriálu
4. Dlouhodobá diarizace[1] (Longitudinal speaker diarization) řečníka na několika různých nahrávkách

Jelikož je úloha č. 2 pro tuto práci podstatná, její podrobnější popis lze nalézt v sekci 3.4.

Data použitá v prvním ročníku obsahovala přibližně 1600 hodin audio nahrávek z pořadů vysílaných na stanicích BBC1, BBC2, BBC3 a BBC4 mezi 1. dubnem a 19. květnem 2008. Nahrávky byly rozděleny do několika sad s různým určením (vývoj, testování). Podrobnější informace lze nalézt v [4].

[1] Rozdělení nahrávek na segmenty podle identity řečníka

3.2 MGB-2

MGB-2 [1] obsahovala dvě úlohy:

1. Automatický přepis řeči na text (Text-to-speech transcription)
2. Zarovnání (Alignment) audia k textovému přepisu

Na rozdíl od MGB-1 byly použity nahrávky v arabštině pocházející z vysílání televizní stanice Aljazeera. Data byla posbírána z rozmezí let 2005 až 2015 z 19 různých programů s celkovou délkou více než 1200 hodin. Data obsahují různé arabské dialekty.

3.3 MGB-3

MGB-3 [2] obsahovala následující dvě úlohy:

1. Automatický přepis řeči (Text-to-speech transcription)
2. Identifikace arabských dialektů (Arabic Dialect Identification)

Data byla přebrána z MGB-2, ovšem byly přidány nové nahrávky. Na rozdíl od MGB-1 a MGB-2 nebyla nová data získána z televizního vysílání, nýbrž se jedná o nahrávky z YouTube o celkové délce 16 hodin. Pocházejí z videí různých žánrů: komedie, vaření, móda, sporty nebo vědecké přednášky TED.

3.4 Zarovnání audia k titulkům (MGB-1)

V této úloze dostali účastníci k dispozici titulky z vysílání bez časových informací. Jejich úkolem bylo zarovnat tyto titulky ke zvukovým stopám na úrovni jednotlivých slov, pokud to bylo možné. Televizní titulky se totiž často liší od skutečně vyslovených slov z důvodu lepší pochopitelnosti, parafrázování nebo příliš rychlé řeči. V titulcích tedy mohou chybět nebo naopak přebývat některá slova.

Pro tuto úlohu byla určena následující data: trénovací sada `train.full` (2193 nahrávek o délce 1580 hodin), vývojová sada `dev.full` (47 nahrávek o délce 28 hodin) a evaluační sada `eval.std` (16 nahrávek o délce 11 hodin). [4]

Dodaný skript pro skórování byl vytvořen Thomasem Hainem[2] pro účely MGB Challenge. Skript počítá vážený harmonický průměr známý jako F score (viz 3.5.3) z hodnot recall (viz 3.5.1) a precision (viz 3.5.2). Slovo bylo považováno za správně zarovnané, pokud čas začátku a konce slova spadá do 100 ms velkého okna. Tento skórovací skript je použit i pro účely této práce, podrobné informace k způsobu skórování jsou v sekci 3.5. Účastníci měli dovoleno zahrnout ve výsledném zarovnání pouze ta slova, která se nacházela v dodaném přepisu, ovšem mohli některá slova odstranit. Ze skórování byla odstraněna slova, obsažená v částech, kde se překrývalo více promluv.

3.5 Skórování

Pro vyhodnocování výsledků v oblasti rozpoznávání a klasifikace bývají mimo jiné použity míry známé jako recall, precision a jejich vážený harmonický průměr F score. V této sekci

[2] http://www.dcs.shef.ac.uk/~th/

jsou vysvětleny tyto míry a přiblížen postup, jakým pracuje skript použitý pro skórování výsledků experimentů.

Pokud provádíme binární klasifikaci (přiřazení jedné ze dvou tříd prvkům nějaké množiny) získáváme jednak predikovanou příslušnost (P) a jednak skutečnou příslušnost (R) daného prvku do jedné ze tříd. Řekněme, že tyto dvě třídy označíme např. *positive* a *negative*. Výsledek klasifikace tedy může spadat do jedné ze čtyř kategorií, tak jak ukazuje tabulka 3.1 [12].

	R positive	R negative
P positive	true positive	false positive
P negative	false negative	true negative

Obrázek 3.1: Možné výsledky klasifikace.

Jednotlivé kategorie mají následující význam:

- true positive – hodnota byla korektně klasifikována jako pozitivní
- false positive – hodnota byla nekorektně klasifikována jako pozitivní
- true negative – hodnota byla korektně klasifikována jako negativní
- false negative – hodnota byla nekorektně klasifikována jako negativní

Pomocí těchto kategorií jsou definovány míry recall, precision a F score popsané v následujících sekcích.

3.5.1 Recall

Recall, česky lze přeložit jako výtěžnost [6], je definován jako poměr korektně určených pozitivních hodnot, ke všem pozitivním hodnotám [12]. Můžeme ji tedy spočítat jako

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}}$$

3.5.2 Precision

Precision, česky lze přeložit jako přesnost [6], je definována jako poměr korektně určených pozitivních hodnot ke všem klasifikací získaným hodnotám [12]. Můžeme ji tedy spočítat jako

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}}$$

3.5.3 F score

F score, také známé jako F measure či F1, je definováno jako vážený harmonický průměr přesnosti a vytíženosti [12]. Lze jej spočítat jako

$$\text{F score} = 2 \cdot \frac{\text{precision} \cdot \text{recal}}{\text{precision} + \text{recall}}$$

3.5.4 Skript `score-alignment.py`

Jak již bylo předesláno v sekci 3.4, skript použitý pro skórování v této práci byl vytvořen pro úlohu č. 2 v MGB Challenge 2015 a počítá míru F score pro jednotlivé zarovnávané nahrávky.

Skript je spouštěn následovně:

`score-alignment.py [options] RefCTM SysCTM`,

kde `RefCTM` je referenční zarovnání ve formátu CTM, `SysCMT` je skórované zarovnání ve formátu CTM a `[options]` mohou být:

- `--script ScriptSTM` – referenční přepis
- `--ignoreintervals ignoreUEM` – ignorování slov ve specifikovaných intervalech
- `--match_bound FRAMES` – počet jednotek (o velikosti 10 ms) nalevo a napravo, do kterých může spadat začátek a konec slova, aby bylo považováno za správně zarovnané
- `-h, --help` – pro zobrazení nápovědy (jsou dostupné i další přepínače)

Jelikož míry recall a precision nevyžadují znát počet *true negative*, skript počítá pouze správně zarovnaná slova (*true positive*). Součty *true positive* + *false negative* a *true positive* + *false positive* jsou známy a jedná se o počty slov ve skórovaném a v referenčním zarovnání. Skript umožňuje ignorovat předem specifikované časové intervaly, ve kterých se překrývá více řečníků, a proto jsou zmíněné množiny před výpočty filtrovány.

Kapitola 4

Postup zarovnávání a příprava na experimenty

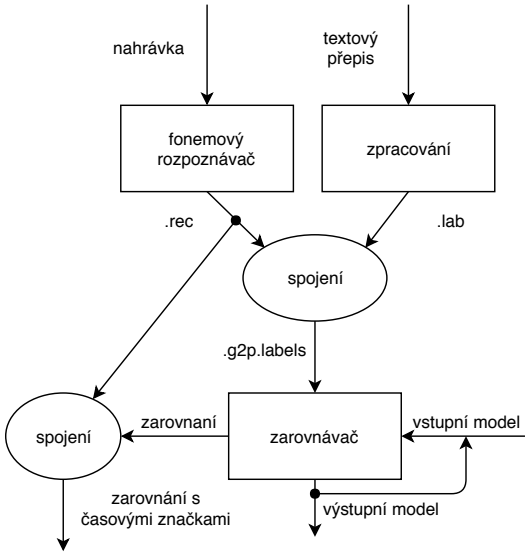
V této kapitole si popíšeme obecný postup zarovnávání, postupné adaptace modelu jedné nahrávky a trénování modelů na více nahrávkách současně. Dále se podíváme na to jaké nástroje byly použity k provedení experimentů a na jakých datech byly tyto experimenty provedeny. Následující kapitola poté popisuje jednotlivé experimenty a jejich průběh.

4.1 Obecný postup zarovnávání

K zarovnávání se používá dodaný skript `g2p_alignment.py`, který byl vytvořen Mirko Hannemannem[1]. Pokud skript spustíme běžným způsobem, který je popsán v sekci 4.4, dojde k natrénování modelu pro zarovnávání na dané nahrávce a poté k samotnému jejímu zarovnání spočítaným modelem.

Blokové schéma procesu zarovnávání je na obrázku 4.1. Vstupními soubory zarovnávacího procesu je nahrávka a její textový přepis.

[1] http://www.fit.vutbr.cz/~ihannema/



Obrázek 4.1: Blokové schéma obecného procesu zarovnání

Nahrávku, která by v obecném případě mohla být i video, je potřeba převést do formátu podporovaného použitým fonémovým rozpoznávačem. Následně je vytvořen její fonémový přepis ve formátu ukázaném v sekci 2.3.2, který obsahuje i informace o časech výskytu jednotlivých fonémů. Tato data jsou uložena v souboru s příponou `.rec`.

Textový přepis může být také v různých formátech. Je třeba jej převést na formát, kdy se každé slovo nachází na jednom řádku. Během převodu je potřeba provést expanzi případných číslovek, či odstranění znaků, se kterými si zarovnávací skript neporadí z důvodu jejich speciálního významu. Jedná se například o znaky levá a pravá závorka nebo středník. Výstupem toho předzpracování je soubor s příponou `.lab`.

Ze souborů s fonémy a se slovy je vytvořen jediný soubor s příponou `.g2p.labels`. Soubor se skládá ze dvou řádků: na prvním jsou všechna slova textového přepisu, která jsou doplněná o značky uvozující a zakončující celou řeč a jednotlivá slova, na druhém řádku se nacházejí všechny fonémy ovšem již bez časových informací, neboť ty nejsou pro samotné zarovnání potřeba. Příklad formátu `.g2p.labels` souboru je následující:

```
<u> T O D A Y <w> W E ' R E <w> R O A M I N G <w> A R O U N D <w> T H E <u>
ay d uw d ey w t r iy l w t eh l ay n hh ih s t aa r k s t ow p iy n s f
```

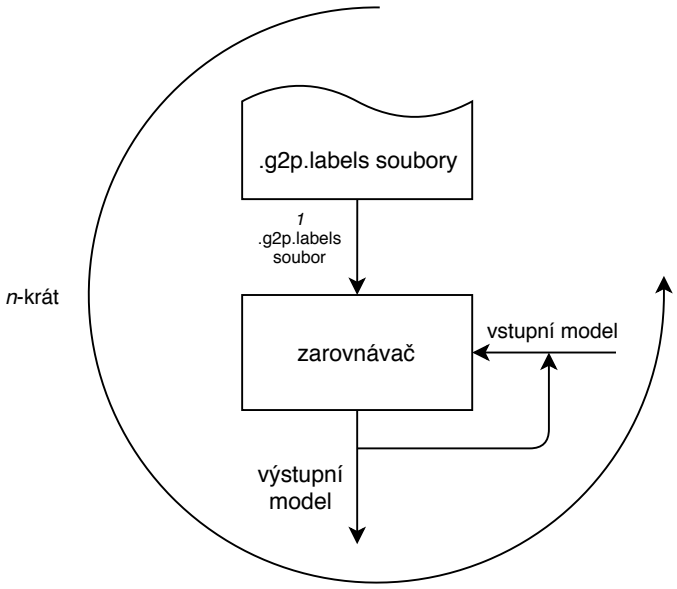
Tento soubor je vstupem samotného zarovnávače. Zarovnávač může pracovat dvěma způsoby: natrénuje na dané nahrávce model a vytvoření podle tohoto modelu zarovnání nebo je mu dodán již natrénovaný model a provede se pouze zarovnání. Výstupem zarovnávače je soubor s podobným formátem jako `.g2p.labels`, obsahuje ovšem navíc spočítané mapování jednotlivých grafémů na fonémy. Tento výstupní soubor samozřejmě opět neobsahuje časové informace a je tedy zkombinován s `.rec` souborem obsahujícím informaci o časování jednotlivých fonémů, čímž je získána informace o časování jednotlivých slov z původního přepisu. Výstup může být dále zpracován do různých formátů pro účely dalšího zpracování (například pro skórování, formát `srt` pro titulky a podobně).

4.2 Adaptace modelu

Nástroj `g2p_alignment.py` umožňuje při správném spuštění s určitými parametry (viz 4.4) provést adaptaci existujícího modelu novými daty získanými z jiné nahrávky. Má to ovšem jistá omezení. Jelikož se nástroj při adaptaci neučí již žádné nové fonémy nebo grafémy, ale pouze upravuje pravděpodobnosti mapování grafémů na fonémy, nesmí nahrávka, kterou chceme existující model adaptovat, obsahovat žádné fonémy nebo grafémy, které nebyly obsaženy v původní nahrávce. V případě, že k tomuto dojde, zahlásí zarovnávač chybu a proces nelze dokončit. V případě fonémů k této chybě téměř nedochází, vzhledem k celkem malému množství různých fonémů, které bývají často všechny obsaženy v každé nahrávce. Textové přepisy na druhou stranu obsahují velké množství různých symbolů a původní textové přepisy nebylo možno použít, podrobnější popis tohoto problému a jeho řešení je v sekci 4.5.1.

4.2.1 Obecný postup adaptace modelu

Blokové schéma procesu adaptace modelu je na obrázku 4.2. Za vstup tohoto procesu můžeme považovat již připravené `.g2p.labels` soubory, jejichž vytvoření by probíhalo stejně jako v případě samotného zarovnávání.



Obrázek 4.2: Blokové schéma procesu adaptace modelu n nahrávkami.

Předpokládejme, že chceme adaptaci provést s n nahrávkami. Adaptace probíhá iteračně. V každé z n iterací vezmeme 1 `.g2p.labels` soubor a předáme jej jako vstup zarovnávače společně s modelem, který adaptujeme. V první iteraci bude tento vstupní model prázdný. Jakmile zarovnávač ukončí svoji činnost, je do souborů původně vstupního modelu zapsán model nový, který se v další iteraci stává opět vstupním modelem. Výstupem zarovnávače je také soubor s provedeným zarovnáním aktuálním modelem, tento soubor je zahazován.

Celý proces byl automatizován shell skriptem popsaným v sekci 4.4. Během adaptace dochází občas k vypuštění grafému `<u>` z modelu a není potom možné dále takový model upravovat nebo s ním zarovnávat z toho důvodu, že tento grafém symbolizuje začátek a

konec řeči v textovém přepisu a je tedy obsažen v každém `.g2p.labels` souboru. Je možné, že se jedná o chybu zarovnávače a problém je řešen kontrolou absence tohoto grafému a případným dodatečným doplněním zpět do modelu. Dále je potřeba po skončení všech adaptací upravit název souboru obsahující ceny přechodů z `*.1.1` na `*.1`, aby bylo možné takovým modelem zarovnávat.

4.3 Trénování modelu na více nahrávkách

Nástroj `g2p_alignment.py` ve verzi, v které byl dodán nepodporuje možnost zadat naráz několik různých nahrávek a natrénovat nad nimi model pro zarovnávání. Skript byl tedy upraven aby tuto funkčnost podporoval. Nová verze se jmenuje `g2p_alignment_train.py`.

4.3.1 Provedené změny

V této sekci krátce popíšeme úpravu, kterou bylo potřeba udělat, aby skript dokázal natrénovat model na více nahrávkách.

Skript po spuštění provádí nejrůznější operace, jednou z nich je zavolání funkce `load_lexicon()`,která má za úkol načíst do paměti `.g2p.labels` soubor a jeho obsah přidat jako příklad pro trénování. V této funkci skript přečte ze souboru první dva řádky a jejich obsah předá další funkci `add_example()` a tím práce se vstupním souborem končí. Podívejme se na útržek kódu provádějící tyto operace:

```
def load_lexicon(lexname, graphones_loaded, lattices_loaded, command):
  # ...
  file = codecs.open(lexname, "r", "utf-8")
  # ...
  g = file.readline().strip().split()
  p = file.readline().strip().split()
  if not lattices_loaded:
    add_example(g, p, graphones_loaded, 0, command == "decode")
  remember_example(g, p, graphones_loaded, 0)
  # ...
```

Důležité bylo rozhodnutí, jakým způsobem více nahrávek přidávat, zda-li všechny uložit do jednoho `.g2p.labels` souboru, kdy vždy každé dva řádky obsahují jednu nahrávku, nebo nechat uživatele zadat adresář obsahující několik `.g2p.labels` souborů. Vzhledem k tomu, že soubory již byly vytvořeny a bylo potřeba jednoduše připravovat data pro experimenty, byla zvolena varianta s adresářem. K tomu bylo potřeba upravit i způsob otevírání souborů. Upravený kód je následující:

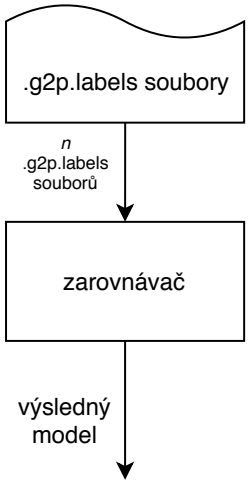
```
def load_lexicon(lexname, graphones_loaded, lattices_loaded, command):
  # ...
  files = []
  if os.path.isdir(lexname):
      print os.listdir(lexname)
      for f in os.listdir(lexname):
          files.append(io.open(lexname + "/" + f, "r", encoding="utf-8"))
  else:
      files.append(io.open(lexname, "r", encoding="utf-8"))
  # ...
```

```
example = 0
for file in files:
    g = file.readline().strip().split()
    p = file.readline().strip().split()
    if not lattices_loaded:
        add_example(g, p, graphones_loaded, example, command == "decode")
    remember_example(g, p, graphones_loaded, example)
    example = example + 1
# ...
```

Skript se nejprve podívá, jestli zadaný soubor je nebo není adresář, a otevře buď jediný soubor, nebo všechny obsažené v daném adresáři, v obou případech jsou objekty obsahující otevřený soubor uloženy do pole. Možnost zadat pouze jediný soubor na vstup je tedy zachována. V rámci přidávání nahrávek je poté toto pole se soubory zpracováno cyklem a z každého souboru jsou přečteny první dva řádky, které jsou uloženy jako příklad pro zarovnávač. Důležité je počítadlo přidaných příkladů (`example`), neboť je potřeba pro správnou funkčnost tuto informaci předat funkcím `add_example()` a `remember_example()`.

4.3.2 Postup trénování

Blokové schéma procesu trénování modelu je na obrázku 4.2. Za vstup tohoto procesu můžeme považovat již připravené `.g2p.labels` soubory, jejichž vytvoření by probíhalo stejně jako v případě samotného zarovnávání nebo adaptace.



Obrázek 4.3: Blokové schéma procesu trénování modelu na n nahrávkách.

Jak můžeme vidět, proces trénování je na rozdíl od adaptace velmi jednoduchý. Namísto n iterací po jedné nahrávce, je možné do zarovnávače naráz předat všech n `.g2p.labels` souborů a ty jsou v jednom spuštění skriptu zpracovány a je vytvořen finální model, který není ani potřeba nijak dále upravovat. Do určité míry je také vyřešen problém s neznámými grafémy a fonémy, neboť jsou načteny ze všech použitých nahrávek naráz. Problém ovšem stále přetrvává v případě, že tímto modelem chceme zarovnávat nahrávky, obsahující další neznámé fonémy nebo grafémy.

4.4 Použité nástroje

Tato sekce obsahuje seznam použitých nástrojů s jejich krátkým popisem. Mimo dále v této sekci zmíněných programů a skriptů byla ještě použita celá řada dalších pomocných a automatizačních skriptů.

SoX (Sound eXchange)[2] je terminálový nástroj pro konvertování formátů zvukových souborů. Byl použit k úpravě vzorkování nahrávek.

phnrec je fonémový rozpoznávač vyvinutý na Fakultě informačních studií Vysokého učení technického v Brně výzkumnou skupinou Speech@FIT[3]. Vychází z disertační práce Petra Schwarze [14]. V základu obsahuje systémy pro anglický, český, maďarský a ruský fonémový přepis, ale je možné použit i vlastní systém.

prepare_labels.sh je dodaný skript provádějící spojení souborů s fonémy a grafémy do jednoho souboru `.g2p.labels`.

g2p_alignment.py je dodaný skript pro G2P zarovnání založený na [5]. Byl vytvořen Mirko Hannemannem. Skript slouží jak k natrénování modelu, tak k samotnému zarovnání.

Použití pro adaptaci modelu:

```
g2p_alignment.py train-align input.g2p.labels output.graphones output.costs
  output.alignemnt beam
```

kde

- `input.g2p.labels` – vstupní soubor s grafémy a fonémy
- `output.graphones`, `output.costs` – výstupní soubory pro natrénovaný model
- `output.alignment` – výstupní soubor s provedeným zarovnáním bez časových informací
- `beam` – maximální relativní vzdálenost mezi pozicí grafému a fonému, nižší hodnota výrazně snižuje čas potřebný pro zarovnání, ale snižuje také přesnost

Použití pro zarovnávání:

```
g2p_alignment.py align input.g2p.labels input.graphones input.costs
  output.alignemnt beam
```

kde

- `input.g2p.labels` – vstupní soubor s grafémy a fonémy
- `input.graphones`, `input.costs` – natrénovaný model
- `output.alignment` – výstupní soubor s provedeným zarovnáním bez časových informací
- `beam` – maximální relativní vzdálenost mezi pozicí grafému a fonému, nižší hodnota výrazně snižuje čas potřebný pro zarovnání, ale snižuje také přesnost

[2] http://sox.sourceforge.net/
[3] http://speech.fit.vutbr.cz/software/phoneme-recognizer-based-long-temporal-context

g2p_alignment_train.py je upravená verze předcházejícího skriptu pro zarovnávání. Byla upravena funkce sloužící pro přidávání příkladů pro trénování modelu. V původní verzi nebylo možné skriptu předat více než jednu nahrávku pro trénování modelu. Zmíněná úprava umožňuje namísto jednoho souboru `input.g2p.labels` zadat celý adresář a v rámci jednoho běhu skriptu je model natrénován na všech příkladech z daného adresáře.

alignment2mlf.py je skript vytvořený Mirko Hannemannem kombinující `.alignment` soubor s `.rec` souborem za účelem přidání časových značek k vypočítanému zarovnání fonémů na grafémy. Výsledek je uložen ve formátu `mlf`.

timings2ctm.sh převádí zarovnání do formátu vhodného pro skórování.

score_alignment.py je skórovací skript vytvořený pro účely MGB Challenge 2015 (viz 3.1). Popis použití skórovacího skriptu se nachází v sekci 3.5

score2rawData.sh převádí výstup score_alignment.py do formátu vhodného pro vytváření grafů. Výstup obsahuje řádky

```
beam precision name score
```

kde `beam` je hodnota použitá při zarovnávání, `precision` je tolerance při skórování, `name` je název nahrávky a `score` je skóre dané nahrávky. Poslední řádek obsahuje celkové F skóre.

adaptNmodel.sh je vyvinutý skript pro adaptaci modelu jedné nahrávky na dalších nahrávkách. Opakovaně spouští zarovnávač nad jednotlivými nahrávkami (včetně jedné původní) a postupně adaptuje natrénovaný model.

Spustění:

```
adaptNmodel.sh directory amount g2p_labels [scores] [reverse]
```

kde

- `directory` – výstupní adresář pro výsledný model
- `amount` – počet, kolik nahrávek má být v procesu použito, musí být menší než počet souborů v `g2p_labels`
- `g2p_labels` – adresář obsahující `.g2p.labels` soubory
- `scores` – soubor se skóre jednotlivých nahrávek, výstup skriptu score2rawData.sh, kde `beam` a `precision` mohou být libovolné a `score` určuje výsledné pořadí nahrávek při adaptaci. Patřičnou úpravou hodnot `score` lze tedy specifikovat i jiné pořadí než jen to vycházející z výsledků skórování. Pokud je parametr vynechán, jsou nahrávky náhodně zamíchány.
- `reverse` – pokud je zadán pátý parametr, je pořadí nahrávek podle `scores` obráceno.

trainNmodel.sh je vyvinutý skript pro trénování modelu na více nahrávkách. Oproti adaptaci modelu není zarovnávač opakovaně spouštěn, nýbrž jsou během jednoho spuštění předány všechny nahrávky naráz.

Spustění:

```
trainNmodel.sh directory amount g2p_labels [scores] [reverse]
```

kde

- `directory` – výstupní adresář pro natrénovaný model
- `amount` – počet, na kolika nahrávkách má být model trénován, musí být menší než počet souborů v `g2p_labels`
- `g2p_labels` – adresář obsahující `.g2p.labels` soubory, ze kterých skript podle určeného pořadí vybere vhodné soubory
- `scores` – soubor se skóre jednotlivých nahrávek, výstup skriptu score2rawData.sh, kde `beam` a `precision` mohou být libovolné a `score` určuje výsledné pořadí nahrávek při trénování. Patřičnou úpravou hodnot `score` lze tedy specifikovat i jiné pořadí než jen to vycházející z výsledků skórování. Pokud je parametr vynechán, jsou nahrávky náhodně zamíchány.
- `reverse` – pokud je zadán pátý parametr, je pořadí nahrávek podle `scores` obráceno.

4.5 Použitá data

Data, která byla pro experimentování dodána pochází z MGB Challenge 2015 (viz 3.1). Jedná se vývojovou sadu určenou pro evaluační úlohu č. 2 (viz sekce 3.4).

4.5.1 Textové přepisy

Textové přepisy byly dodány ve formátu XML. Každý soubor obsahuje několik sad přepisů z nichž byl pro experimenty v této práci vybrán přepis `human_transcript`, tedy člověkem ručně připravený přepis. Je nutné podotknout, že tento přepis nebude nutně odpovídat s naprostou přesností tomu referenčnímu, který byl dodán spolu se skórovacím skriptem. I z tohoto důvodu není teoreticky možné dosáhnout nejvyššího F score. Přepisy také mohou obsahovat zástupné znaky za slova, kterým přepisovatel nerozuměl. Jak již bylo zmíněno dříve, každý přepis nutně prochází drobnými úpravami před tím, než je předán na vstup zarovnávače. Tuto sadu přepisů budeme označovat podle jejího původního označení a to `human_transcript`.

Pro účely adaptace modelu více nahrávkami bylo kromě úprav zmíněných v sekci 4.1 nutno provést i další zásahy. Jelikož byly přepisy vytvářeny člověkem, došlo na několika místech k překlepům, kdy některá slova obsahovala namísto velkých písmen i písmena malá, což do zarovnávání zanášelo další grafémy, které následně mohly být pro adaptovaný model neznámé. I kdyby se podobná chyba vyskytovala ve všech použitých nahrávkách a nejednalo se o neznámý grafém, zanáší to do modelu chybu v podobě dvou grafémů se stejným významem, které by měly být mapovány na stejný foném. Jak bylo zmíněno v předchozím odstavci, pokud přepisovatel například nerozuměl některému slovu, nahradil jej otazníky, tyto otazníky byly z přepisů odstraněny, stejně jako další speciální znaky, které přepisovatelé používali. Výsledkem je tedy v textu chybějící část promluvy, se kterou si musí

zarovnávač poradit, namísto snahy mapovat některé fonémy nesprávně na speciální znaky. Výsledná sada grafémových přepisů obsahuje tedy pouze znaky anglické abecedy A-Z a z toho důvodu budeme tuto sadu přepisů označovat jako `human_transcript_az`.

Pro účely srovnání byl dále z dodaných referenčních přepisů pro skórování vygenerován třetí grafémový přepis, který budeme značit jako `reference_transcript`.

Tabulka 4.1 shrnuje výčet použitých grafémových přepisů včetně jejich možného použití pro jednotlivé sady experimentů.

Označení[1]	**Způsob získání**	**Možné použití**[2]
`human_transcript`	dodaný spolu s daty	I
`human_transcript_az`	`human_transcript` pouze s písmeny A-Z	I, II, III
`reference_transcript`	z referenčních přepisů pro skórování	I, II, III

[1] označení v rámci této práce

[2] I – experimenty nad vlastním modelem; II – adaptace modelu více nahrávkami; III – trénování modelu na více nahrávkách

Tabulka 4.1: Použité grafémové přepisy.

V rámci experimentu 5.1.5 bylo provedeno srovnání výsledků zarovnání s těmito třemi různými grafémovými přepisy. Jelikož nebylo možné použít původní `human_transcript` pro adaptaci modelu, byl tento experiment proveden pouze nad vlastními modely jednotlivých nahrávek.

4.5.2 Fonémové přepisy

Fonémové přepisy použité v experimentech byly vytvořeny z dodaných nahrávek ve formátu wav pomocí fonémového rozpoznávače phnrec. K rozpoznávání fonémů byl použit nástroj phnrec (sekce 4.4) s různými systémy pro rozpoznávání. Při použití některých systémů bylo potřeba nahrávky převzorkovat na odpovídající vzorkovací frekvenci, k tomu byl použit nástroj SoX.

Označení[1]	**Jazyk**
`eng`	angličtina
`eng_advanced`	angličtina
`hun`	maďarština
`cze_advanced`	čeština

[1] označení v rámci této práce

Tabulka 4.2: Použité fonémové přepisy.

Fonémové přepisy použité v experimentech jsou uvedeny v tabulce 4.2. Základní anglický fonémový přepis vytvořený nástrojem phnrec pomocí volně dostupného systému pro rozpoznávání, který byl natrénován na 20 hodinách anglické řeči pocházející z databáze TIMIT[4]. Tento přepis budeme značit jednoduše `eng`. Dalším použitým přepisem je opět anglický přepis, který byl dodán výzkumnou skupinou Speech@FIT[5] a byl trénován na

[4] https://catalog.ldc.upenn.edu/LDC93S1

[5] http://speech.fit.vutbr.cz/

2000 hodinách anglické řeči z databází Fisher[6] a CALLHOME[7]. Tento kvalitnější přepis budeme označovat jako `eng_advanced`. Třetím přepisem je maďarský fonémový přepis vytvořený nástrojem phnrec s volně přístupným systémem pro maďarštinu, který byl trénován na maďarské databázi SpeechDat-E[8] obsahující nahrávky 1000 různých mluvčích. Přepis bude značen jako `hun`. Posledním přepisem je český fonémový přepis, který byl opět dodán skupinou Speech@FIT a byl vytvořený systémem trénovaným na 300 hodinách řeči. Jelikož je s nástrojem phnrec dostupný i méně kvalitní český systém, budeme tento přepis značit `cze_advanced` po vzoru přepisů anglických.

[6] https://catalog.ldc.upenn.edu/LDC2004S13 a https://catalog.ldc.upenn.edu/LDC2005S13
[7] https://catalog.ldc.upenn.edu/LDC97S42
[8] http://www.fee.vutbr.cz/SPEECHDAT-E/sample/hungarian.html

Kapitola 5

Experimenty a jejich průběh

Tato kapitola se zabývá jednotlivými provedenými experimenty a jejich průběhem. Je rozdělena do tří hlavních sekcí, které odpovídají třem sadám experimentů: v první sadě experimentů (sekce 5.1) je zarovnávání prováděno pouze s modelem vytvořeným na dané nahrávce, v druhé sadě experimentů (sekce 5.2) probíhá zarovnání s modely, vytvořenými postupnou adaptací modelu jedné nahrávky dalšími nahrávkami. Třetí sada experimentů (sekce 5.3) používá k zarovnávání model natrénovaný nad více nahrávkami najednou. Poslední sekce této kapitoly provádí shrnutí výsledků experimentů.

5.1 Experimenty nad vlastním modelem

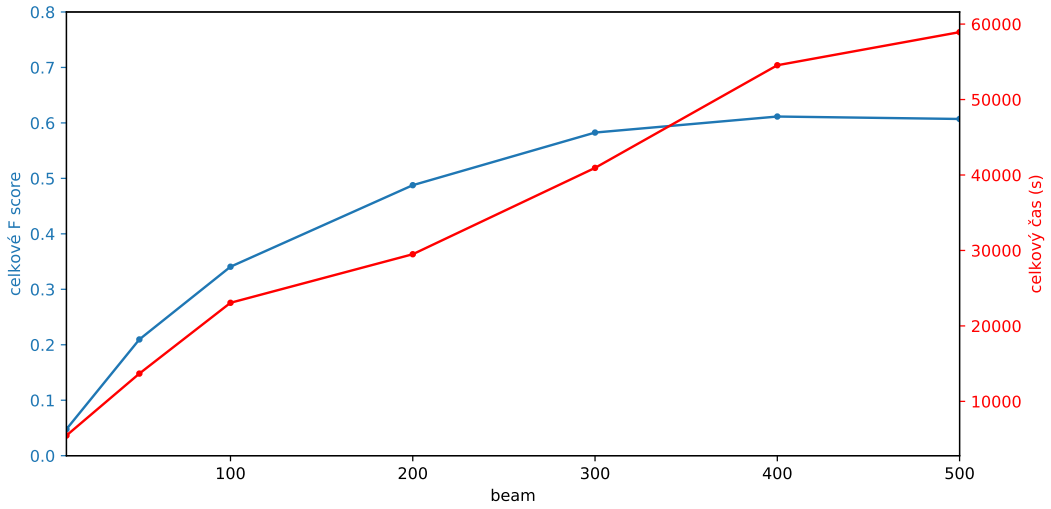
V této sekci jsou popsány experimenty, které byly provedeny s nahrávkami s použitím modelů nad nimi natrénovanými. Experimenty si kladou za cíl zmapovat nahrávky, které budou problémové, zjistit k jak velkým chybám v zarovnání dochází, jaké omezení na šířku vyhledávání je optimální a jak velký vliv na zarovnání má fonémový přepis.

Pro všechny tyto experimenty byly použity textové přepisy `human_transcript` (se zmíněnými drobnými úpravami nutnými pro funkčnost zarovnávače).

5.1.1 Vliv omezení šířky vyhledávání na kvalitu zarovnání

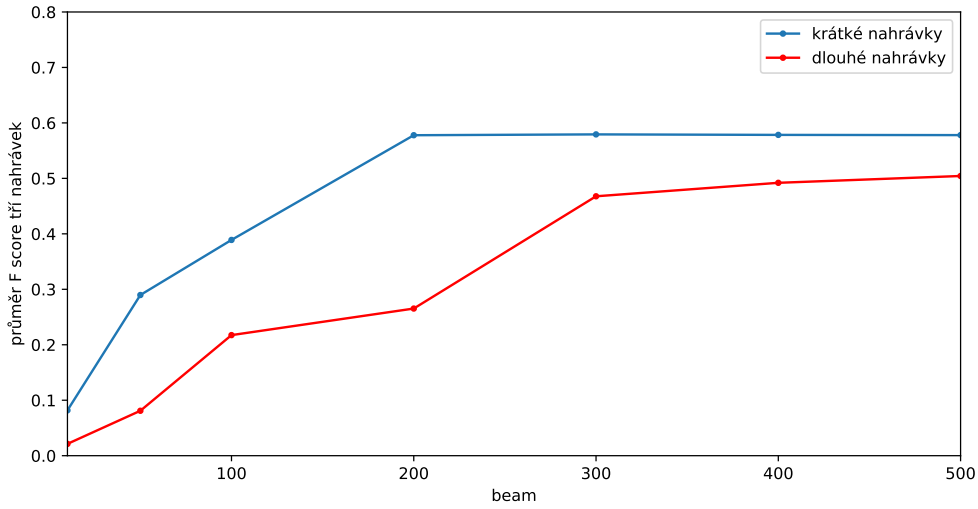
Tento experiment spočívá v provedení zarovnání s různými hodnotami parametru `beam` zarovnávacího skriptu `g2p_alignment.py`. Tento parametr ovlivňuje maximální relativní vzdálenost mezi pozicí grafému a fonému, čímž při nižších hodnotách výrazně zkracuje dobu potřebnou pro zarovnávání, ovšem kvalita zarovnání se snižuje.

V tomto experimentu byly použity fonémové přepisy `eng_advanced`. Skórování pobíhalo s přesností na 100 ms. Všechna zarovnání byla provedena na stejném serveru ve stejných podmínkách. Čas běhu byl změřen jako součet hodnot `user` z výstupu programu `time` spuštěné nad každým během zarovnávacího skriptu pro jednotlivé nahrávky. Hodnoty byly zaokrouhleny na celá čísla.



Obrázek 5.1: Celkové F score pro celou testovací sadu v závislosti na hodnotě parametru `beam` (levá osa y, modrá barva) a celkový čas (v sekundách) potřebný pro zarovnání celé testovací sady v závislosti na hodnotě parametru `beam` (pravá osa y, červená barva). Skórováno s přesností na 100 ms.

Celková hodnota F score pro celou testovací sadu pro různé hodnoty parametry `beam` je zobrazena na grafu 5.1. Na stejném obrázku se nachází i závislost času potřebného k zarovnání celé sady na hodnotě `beam`. Jak můžeme vidět, zlepšení F score je výrazné pro hodnoty parametru do hodnoty 300. Rozdíl mezi hodnotami 300 a 400 již tak razantní není, přičemž čas potřebný pro zarovnání je ovšem výrazně vyšší. Pro hodnotu 500 F score dokonce mírně pokleslo, zatímco potřebný čas stále roste. Zdá se tedy, že již došlo k saturaci a dalšího zlepšení s vyšší hodnotou `beam` již nedojde. Optimální hodnota parametru `beam` pro všechny budoucí experimenty bude 300, jako rozumný poměr mezi kvalitou zarovnání a časem potřebným pro dokončení.



Obrázek 5.2: Průměrné F score pro tři krátké a tři dlouhé nahrávky v závislosti na hodnotě parametru `beam`. Skórováno s přesností na 100 ms.

Podívejme se ještě na vývoj průměrného F score v závislosti na `beam` pro tři krátké a tři dlouhé nahrávky na grafu 5.2. Vidíme, že pro krátké nahrávky dochází k saturaci mnohem dříve a již od hodnoty `beam` 200 dochází k poklesu průměrného F score. U dlouhých nahrávek je zlepšení pozvolnější a i mezi hodnotami 400 a 500 stále získáváme lepší výsledky. Ukazuje se tedy, že u krátkých nahrávek je příliš velká povolená vzdálenost mezi grafémem a fonémem na škodu, neboť může dojít k přílišnému rozhození sekvencí fonémů a grafémů, které nemusí být v rámci krátké nahrávky napraveny. V případě dlouhých nahrávek je prostoru pro případnou nápravu více a k tomuto problému dochází až při vyšších vzdálenostech.

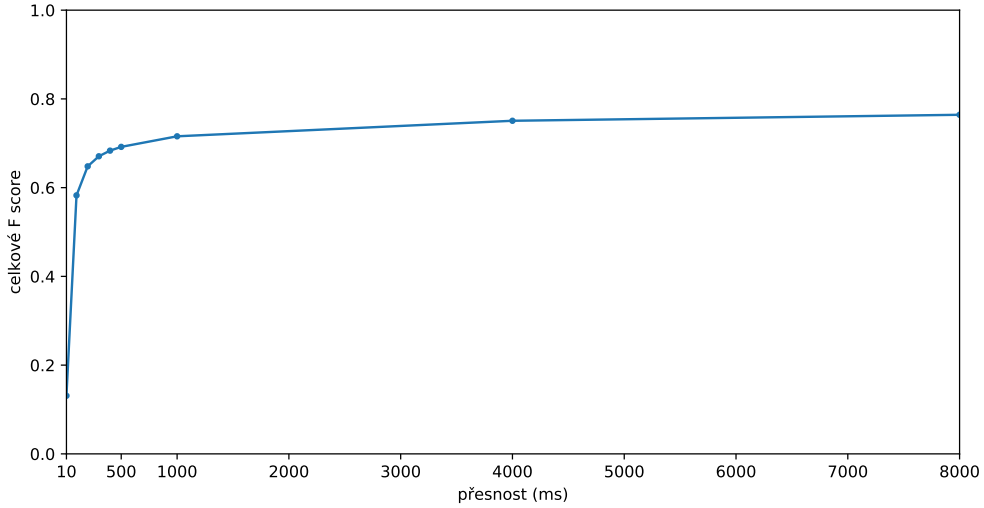
5.1.2 Skórování s různou přesností

Účel tohoto experimentu je zjistit, jakých hodnot F score dosáhnou zarovnané nahrávky, pokud budeme snižovat přesnost skórování (tj. zvyšovat přípustnou chybu). Nastavení přesnosti skórování se provádí přepínačem `--match_bound` skórovacího skriptu (viz sekce 4.4). Větší hodnota tohoto parametru znamená nižší přesnost, ale vyšší F score, neboť zvětšujeme okno kolem referenčního času začátku a konce slova, do kterého se skorované zarovnání musí vlézt, aby bylo slovo považováno za správně zarovnané.

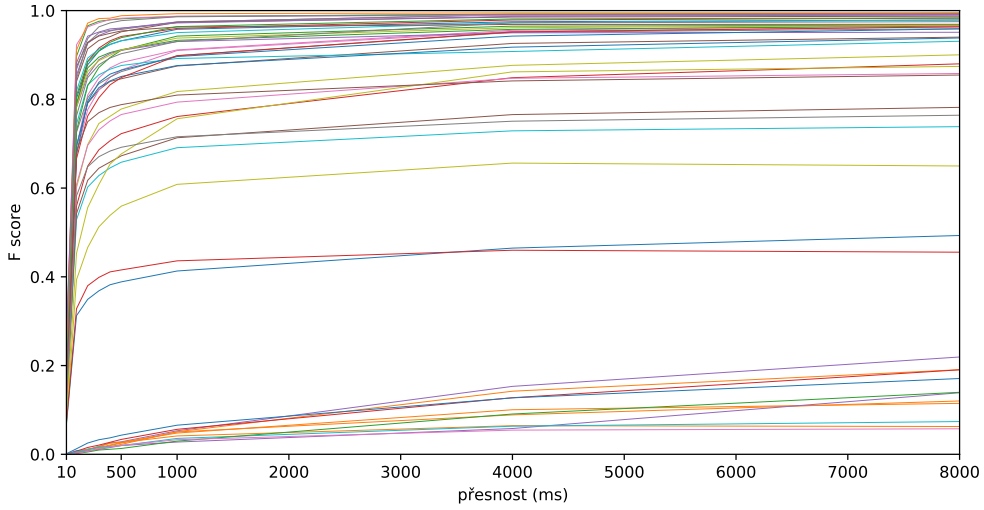
Skórováno bylo zarovnání fonémových přepisů `eng_advanced` s hodnotou parametru `beam` nastavenou na 300. Skórování proběhlo s hodnotami `match_bound` 10, 20, 30, 40, 50, 100, 400 a 800, tedy s přesností 100 ms, 200 ms, 300 ms, 400 ms, 500 ms, 1 s, 4 s a 8 s.

Vývoje celkového F score a F score pro každou nahrávku jsou na grafech 5.3 a 5.4. Můžeme vidět znatelný rozdíl pro přesnosti od 100 ms do 1 s, ovšem pro vyšší hodnoty není zlepšení pro většinu nahrávek již tak dramatické. To znamená, že zbytek chyb v zarovnání není v intervalu od 1 s do 8 s, ale vyšší. Pro některé nahrávky nedosahuje F score hodnoty 0.20 ani při přesnosti 8 s, drtivá většina chyb v těchto nahrávkách tedy bude mnohem vyšší než 8 s. Jedná se převážně o nahrávky ze sportovních pořadů s reportážemi obsahujícími hluk v pozadí a také pořady s vysokým výskytem hudby. Naopak zpravodajské

pořady dosahují vynikajících výsledků i s vysokou přesností. Nejvyšší dosažená hodnota F score pro celou sadu byla 0.7642 s přesností 8 s. Je ovšem důležité poznamenat, že nízké F score nemusí být nutně způsobeno pouze velmi špatným zarovnáním, neboť skórovací skript označí slovo za chybné i v případě, že neodpovídá naprosto přesně slovu referenčnímu. Taková neodpovídající slova jsou považována skriptem za přidaná a správné slovo naopak za chybějící.



Obrázek 5.3: Celkové F score pro celou testovací sadu v závislosti na hodnotě parametru `match_bound`

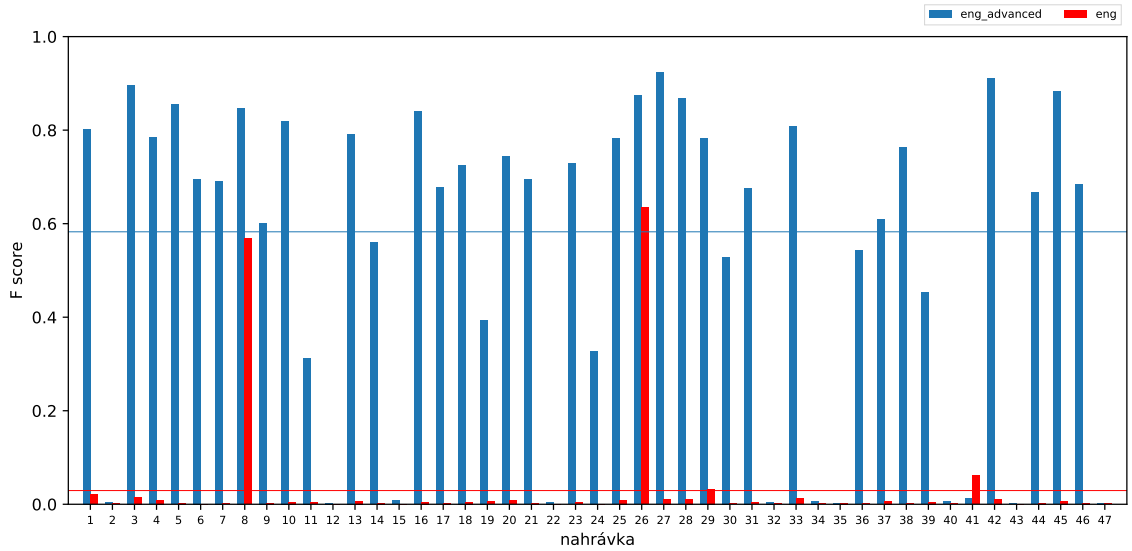


Obrázek 5.4: F score pro jednotlivé nahrávky v závislosti na hodnotě parametru `match_bound`

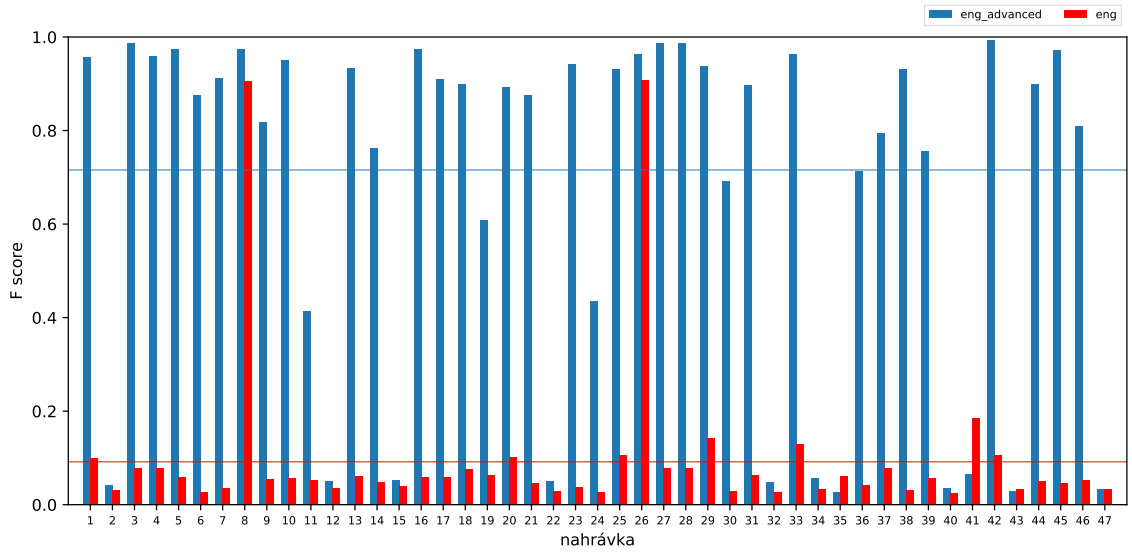
5.1.3 Vliv kvality fonémového přepisu na výsledek zarovnání

V předchozích experimentech jsme používali kvalitní fonémové přepisy `end_advanced`, nyní se podíváme, jak se hodnota F score změní, pokud provedeme zarovnání s méně kvalitními přepisy, konkrétně se sadou, která je v této práci značena pouze `eng`.

Hodnota parametru `beam` byla opět nastavena na hodnotu 300. Graf výsledků skórování pro přesnost na 100 ms a 1 s vidíme na grafech 5.5 a 5.6. Rozdíl je oproti zarovnání z předchozích experimentů znatelný a ukazuje, že kvalita fonémového přepisu zásadním způsobem ovlivňuje výslednou kvalitu zarovnání. Můžeme vidět, že u dvou nahrávek je F score i u nekvalitního přepisu vysoké, jedná se o nahrávky s poměrně čistým zvukem a srozumitelnou řečí bez většího hluku nebo hudby v pozadí. Dále v případě nahrávky č. 41 s přesností na 100 ms a také v případě nahrávky č. 35 s přesností na 1 s můžeme vidět, že zarovnání s nekvalitním přepisem má dokonce lepší výsledek než to s přepisem kvalitním. Jedná se o nahrávku ze sportovního utkání, obsahující tleskání a výkřiky diváků v pozadí, v druhé nahrávce se často vyskytuje smích publika a občasná hudba, dále jeden z řečníků mluví celkem nesrozumitelně. Je tedy možné, že i když byl systém použitý pro fonémové přepisy `eng` trénován na řádově méně hodinách řeči než `eng_advanced`, dokáže si lépe poradit s některými pro rozpoznávání nevhodnými nahrávkami.



Obrázek 5.5: Graf F score jednotlivých nahrávek při použití kvalitních (modrá) a nekvalitních (červená) fonémových přepisů s přesností zarovnání 100 ms. Horizontální čáry reprezentují celkové F score.

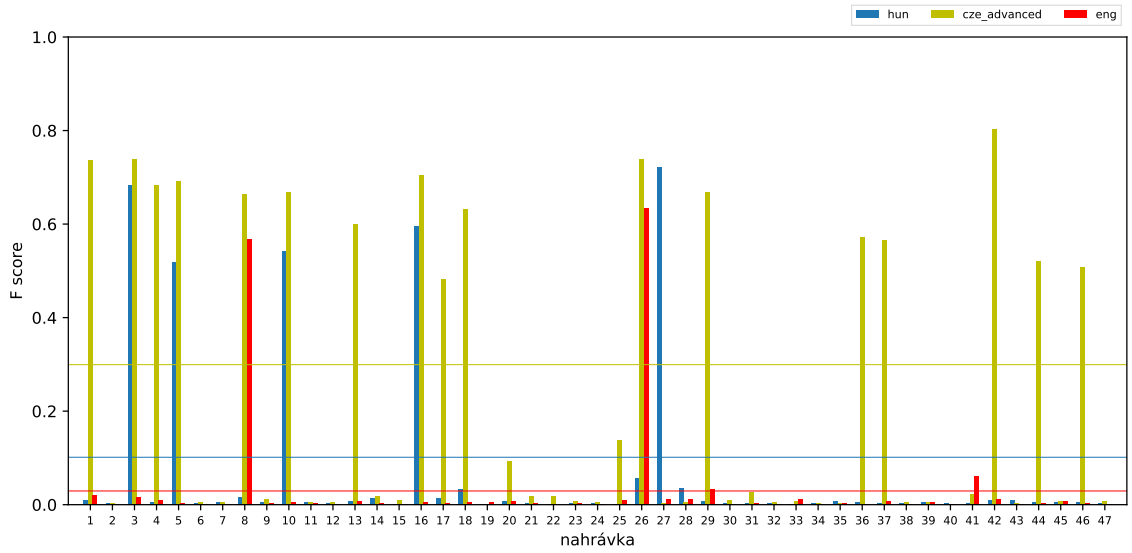


Obrázek 5.6: Graf F score jednotlivých nahrávek při použití kvalitních (modrá) a nekvalitních (červená) fonémových přepisů s přesností zarovnání 1 s. Horizontální čáry reprezentují celkové F score.

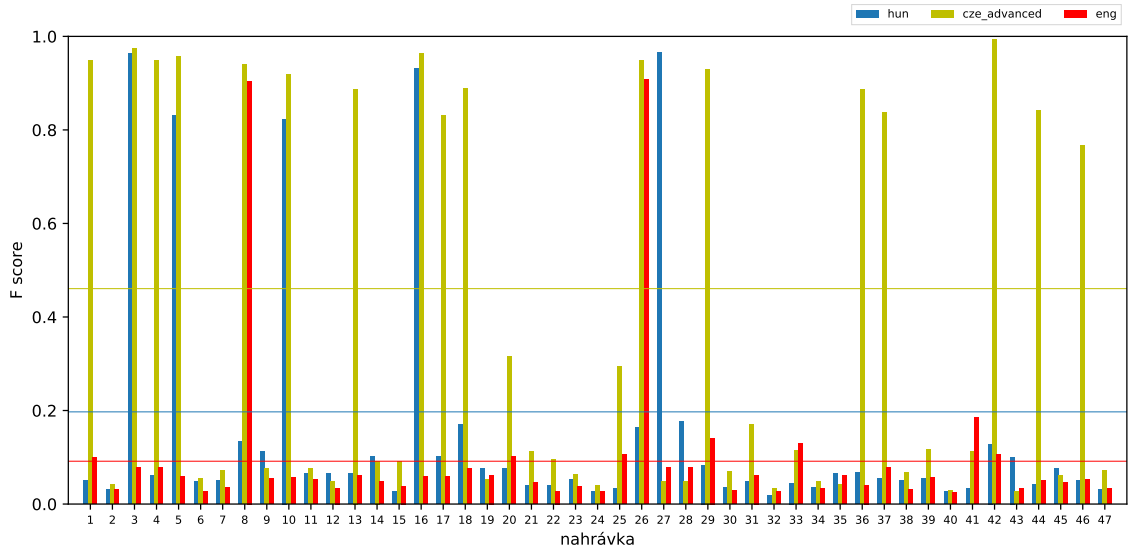
5.1.4 Použití cizojazyčného systému fonémového rozpoznávače pro zarovnání anglické řeči

V tomto experimentu zjistíme, jak kvalitního zarovnání můžeme dosáhnout, pokud nepoužijeme anglický systém pro fonémový rozpoznávač, nýbrž systém maďarský a český. Jedná se tedy o fonémové přepisy, které jsou v této práci značeny jako `hun` a `cze_advanced`. Pro porovnání bylo použito i zarovnání s triviálním fonémovým přepisem `eng`.

Zarovnání opět proběhla s hodnotou parametru `beam` nastavenou na 300. Výsledky pro přesnost skórování 100 ms a 1 s jsou zobrazeny na grafech 5.7 a 5.8. Výsledky ukazují, že i když systém pro fonémový přepis neodpovídal jazyku na nahrávkách, lze v případě maďarského systému v některých případech vidět, že si systém uměl poradit s některými nahrávkami mnohem lépe než triviální anglický systém. Pro dodané české přepisy vytvořené kvalitním systémem k tomuto jevu dochází ještě mnohem častěji. Toto chování může být vysvětleno tak, že samotnému zarovnávači nezáleží na tom jaké fonémy ve skutečnosti zarovnává. Pokud dokáže fonémový rozpoznávač konzistentně rozpoznávat stejné reálné (tedy anglické) fonémy vždy jako ten stejný maďarský či český foném, jedná se v ideálním případě pouze o prostou záměnu symbolů reprezentujících jednotlivé fonémy. V praxi to tak ovšem není a proto nejsou výsledky tak kvalitní jako v případě přepisu `eng_advanced`.



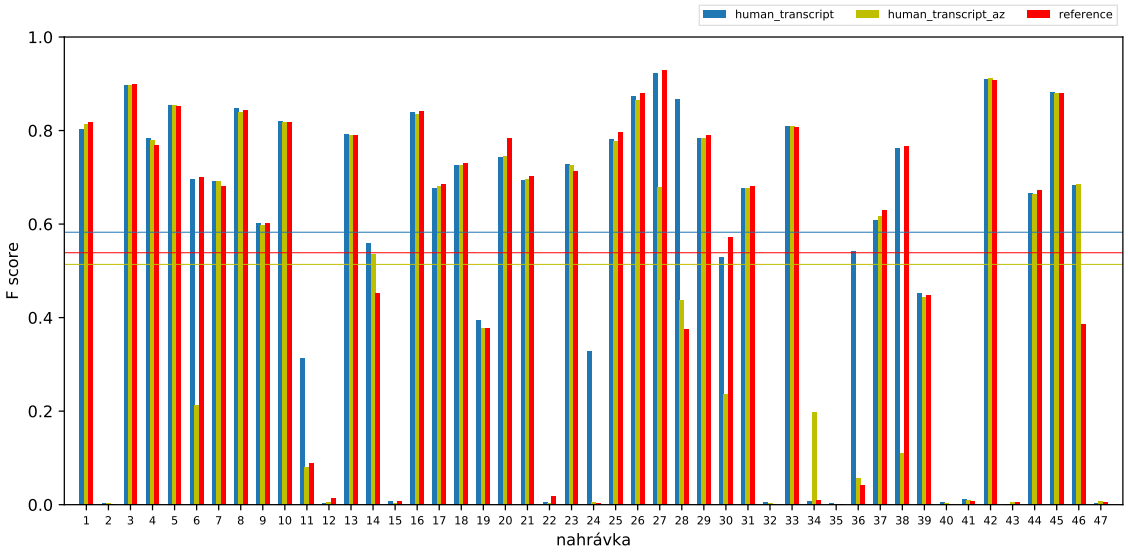
Obrázek 5.7: Graf F score jednotlivých nahrávek při použití maďarských (modrá), českých (žlutá) a nekvalitních anglických (červená) fonémových přepisů s přesností zarovnání 100 ms. Horizontální čára reprezentuje celkové F score.



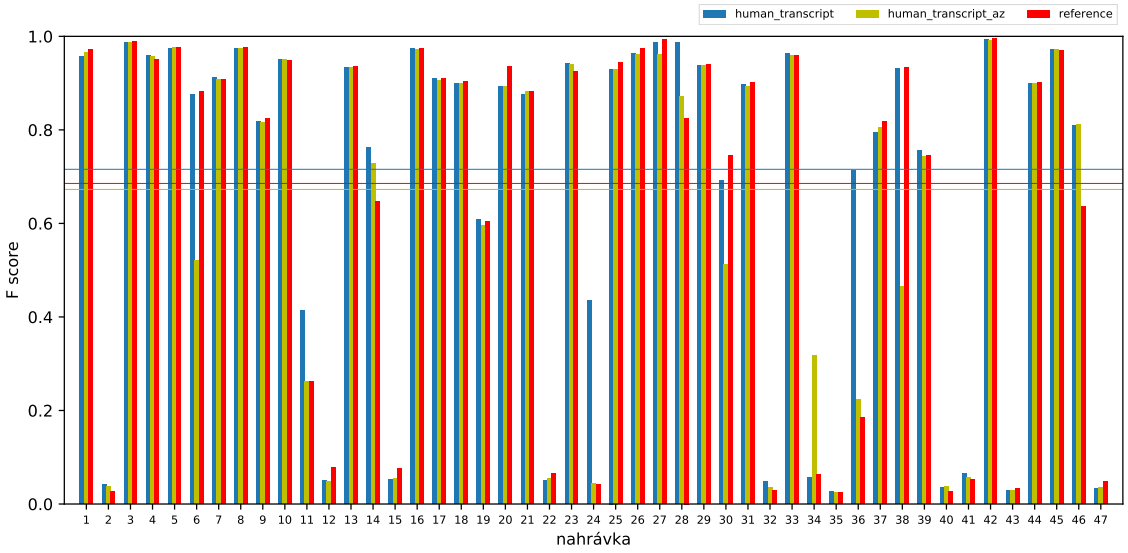
Obrázek 5.8: Graf F score jednotlivých nahrávek při použití maďarských (modrá), českých (žlutá) a nekvalitních anglických (červená) přepisů s přesností zarovnání 1 s. Horizontální čára reprezentuje celkové F score.

5.1.5 Srovnání různých grafémových přepisů

V tomto experimentu byla provedena tři různá zarovnání celé sady nahrávek, každé s jiným textovým přepisem. Zarovnání byla provedena s fonémovým přepisem `eng_advanced` a hodnotou beam nastavenou na 300.



Obrázek 5.9: Graf F score jednotlivých nahrávek s použitím textových přepisů `human_transcript` (modrá), `human_transcript_az` (žlutá), `reference_transcript` (červená). Skórováno s přesností na 100 ms. Horizontální čáry reprezentují celkové F score.



Obrázek 5.10: Graf F score jednotlivých nahrávek s použitím textových přepisů `human_transcript` (modrá), `human_transcript_az` (žlutá), `reference_transcript` (červená). Skórováno s přesností na 1 s. Horizontální čáry reprezentují celkové F score.

Výsledky pro přesnost skórování na 100 ms a 1 s jsou na grafech 5.9 a 5.10. Překvapivě se ukázalo, že původní textový přepis (`human_transcript`) dosahuje nejvyšších hodnot F score. Původní předpoklad byl takový, že odstranění nadbytečných speciálních znaků, které se v mluveném projevu neobjevují a byly přidány pouze přepisovatelem z různých důvodů, by měl zvýšit výsledné F score kvůli vyšší čistotě přepisu. Zdá se ovšem, že si zarovnávač špatně poradí s chybějícími částmi v přepisu a časové značky v okolních slo-

vech jsou tím ovlivněny, což má za následek pokles úspěšnosti zarovnání. Jelikož je model trénován pouze na konkrétní zarovnávané nahrávce, tak existence speciálních znaků jako grafémů v modelu příliš nevadí, ovšem pokud bychom chtěli model používat na další nahrávky, které by také obsahovaly speciální znaky (zejména odlišné neznámé znaky), tak by nebylo vůbec možné tímto modelem nahrávky zarovnat. Tento problém se ostatně vyskytl v experimentech s adaptací modelu na více nahrávkách, kvůli kterým byl právě přepis `human_transcipt_az` vytvořen. I když tento upravený přepis vykazuje horší výsledky než ten původní, bude pro následující experimenty dostačující a umožní vůbec jejich provedení.

Překvapivý je i výsledek referenčního přepisu (`reference_transcript`). Tento přepis nebyl použit v žádných dalších experimentech, neboť cílem práce nebylo dosáhnout všemi možnými způsoby co nejvyššího F score, ale zjistit jak si zarovnávač stojí s reálnými ručně vytvářenými přepisy, které nejsou dokonalé. Zajímavé ovšem je, že tento přepis dosáhl horších výsledků než původní přepis i přes to, že při skórování by mělo docházet k minimálnímu počtu výskytu nesprávných slov (malé procento bude odlišné z důvodu úprav nutných pro správné fungování zarovnávače). Pátrání pro příčině začalo u nahrávky č. 34, kde dosahuje `human_transcipt` výrazně lepších výsledků, než referenční přepis. Podle dat ze skórovacího skriptu a porovnání obou přepisů dané nahrávky se zdá, že lepších výsledků bylo dosaženo zejména kvůli krátké pasáži v nahrávce, která obsahuje krátké útržky řeči oddělené dlouhými pauzami se zvukem na pozadí. V referenčním přepisu nejsou tato hluchá místa nijak vyznačena, zatímco v ručně vytvořeném ano. Zdá se, že tato skutečnost má pozitivní vliv na pozdější pasáž nahrávky, kde jsou přepisy stejné, ovšem v zarovnání s manuálním přepisem došlo k velkému počtu časových shod. Podobný jev se vyskytuje i u dalších nahrávek. Opět se ukazuje, že zarovnávač má problém, pokud v textovém přepisu chybí něco, na co by mohl některé fonémy namapovat. Do jisté míry na to v tomto případě má asi vliv i fonémový rozpoznávač, který mohl některé zvuky na pozadí vyhodnotit jako řeč a přidal neexistující fonémy.

Zajímavá je také nahrávka č. 46, kde manuálně vytvořené přepisy dosahují dobrého výsledku, zatímco referenční přepis výsledku o poznání horšího. Zde se nemůže jednat o problém způsobený místy bez řeči, které by byly vyznačeny speciálními znaky, neboť ty neobsahuje ani přepis `human_transcipt_az`. Nejčastějším rozdílem mezi původním a referenčním přepisem se zdá být výskyt poznámek informujících o hudbě na pozadí, explozích apod., které jsou obvyklé v SDH titulcích (subtitles for deaf or hard-hearing). Ve výsledku se jedná o stejný problém, který bude způsobený nedokonalostí fonémového rozpoznávače a snahou zarovnávače nadbytečným fonémům přiřadit některé grafémy.

5.2 Experimenty s adaptovaným modelem

V této sekci budou popsány experimenty, které byly provedeny s nástroje `g2p_alignment.py` s jeho vestavěnou funkcionalitou adaptovat již existující model další nahrávkou tak, jak bylo popsáno v sekci 4.2. Cílem experimentů je zjistit, jak je možné vylepšit F score zarovnání, pokud model jedné nahrávky přizpůsobíme dalšími a poté tímto modelem zarovnáme celou sadu. Důležitou informací je také optimální počet nahrávek dosahující nejlepších výsledků.

V celé této sadě experimentů je používán textový přepis `human_transcript_az`.

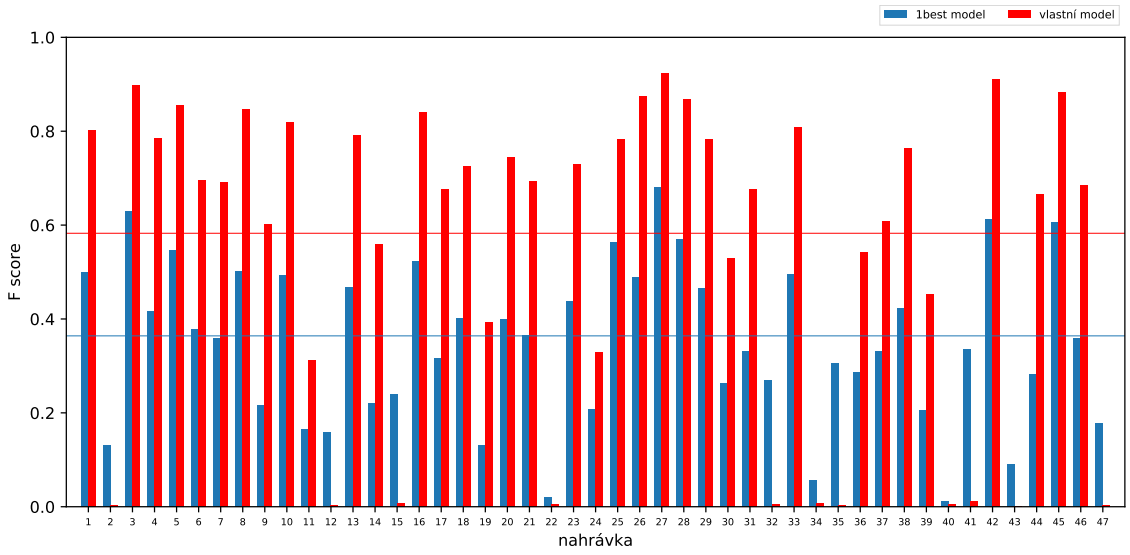
5.2.1 Zarovnání modelem nejlepší nahrávky

V předchozí sadě experimentů byly vždy nahrávky zarovnávány modelem, který byl natrénován na nahrávce samotné. Proto v tomto experimentu ukážeme, jaké výsledky získáme pokud celou sadu zarovnáme jediným modelem, který byl natrénován na jedné nahrávce.

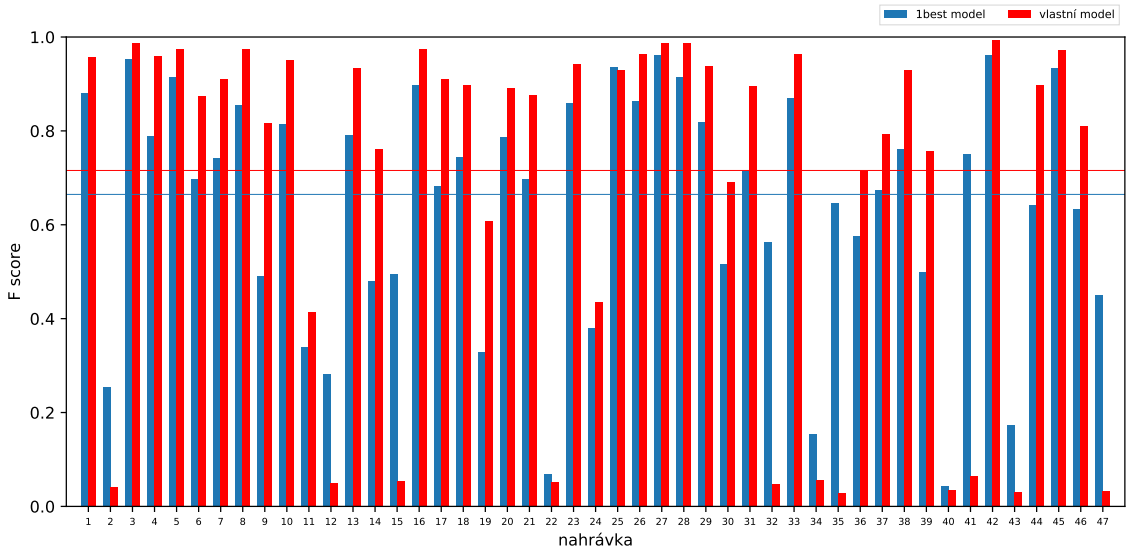
Nahrávka č. 27, jejíž model byl zvolen, dosáhla v experimentu 5.1.3 nejvyššího F score pro fonémový přepis `eng_advanced`. Tento model budeme značit `1best model`. Byl samozřejmě použit stejný fonémový přepis pro zarovnání v tomto experimentu. Hodnota `beam` byla nastavena na 300. Pro srovnání výsledků použijeme výsledky ze zmíněného experimentu. Nebyl použit model z původního experimentu, ale byl znovu natrénován s textovými přepisy, které jsou využívány v této sadě experimentů.

Výsledky experimentu jsou na grafech 5.11 a 5.12. Jak vidíme celkové F score je pro přesnost na 100 ms téměř o polovinu horší, než v případě zarovnání s vlastními modely. Pro přesnost na 1 s je celkové F score tolerovatelné. Co je ovšem zajímavé, je to, že F score jednotlivých nahrávek je sice nižší pro nahrávky, které v předchozím experimentu dopadly dobře, nicméně, u všech nahrávek, které dopadly velmi špatně, je F score nyní minimálně dvakrát lepší (pro nahrávku č. 40, u ostatních podstatně lepší). Zdá se tedy, že zarovnávač nebyl schopný spočítat kvalitní model z těchto velmi špatných nahrávek a pokud použijeme kvalitnější model, je přeci jen možné dosáhnout relativně dobrých výsledků.

Vzhledem k tomu, že používáme z pohledu výsledků zarovnávání méně kvalitní textové přepisy (jak bylo ukázáno v experimentu 5.1.5), nedosáhla ani nahrávka, jejíž model jsme použili (nahrávka č. 27) stejného výsledku jako v předchozím experimentu. V dalších experimentech se pokusíme F score nahrávek vylepšit za pomoci adaptování modelů.



Obrázek 5.11: Graf F score jednotlivých nahrávek s anglickým fonémovým přepisem zarovnaných modelem nejlepší nahrávky (modrá) a vlastními modely jednotlivých nahrávek (červená). Skórováno s přesností na 100 ms. Horizontální čáry reprezentují celkové F score.



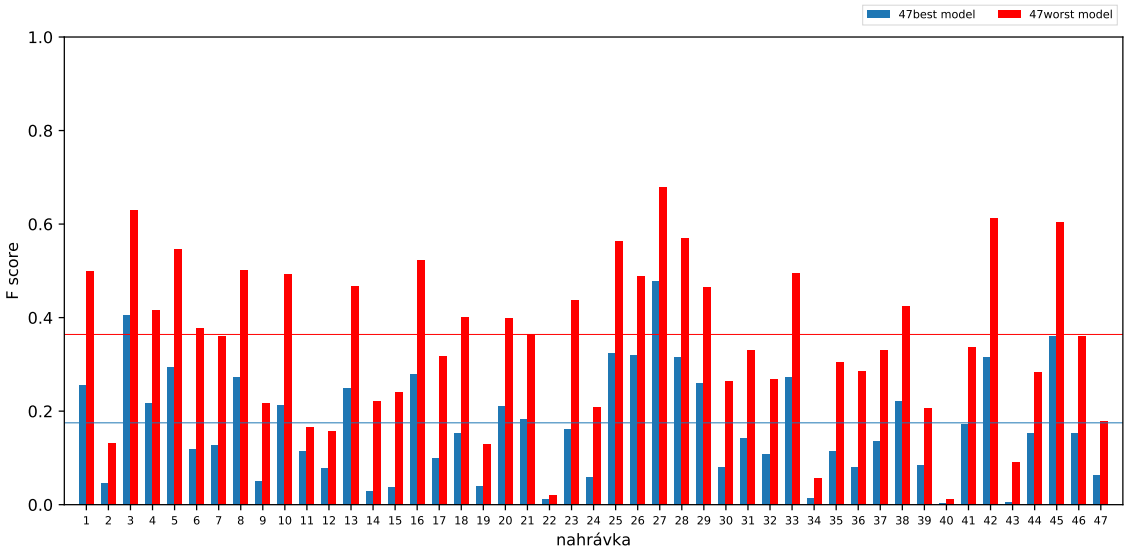
Obrázek 5.12: Graf F score jednotlivých nahrávek s anglickým fonémovým přepisem zarovnaných modelem nejlepší nahrávky (modrá) a vlastními modely jednotlivých nahrávek (červená). Skórováno s přesností na 1 s. Horizontální čáry reprezentují celkové F score.

5.2.2 Zarovnání modelem adaptovaným n nahrávkami

V tomto experimentu provedeme sérii adaptací modelů a následných zarovnání těmito modely za účelem nalezení optimálního počtu nahrávek a způsobu, jakým je postupně používat. Podíváme se také, jaké zlepšení výsledků adaptace modelu přinese pro zarovnání s fonémovými přepisy získanými cizojazyčným systémem pro rozpoznávání.

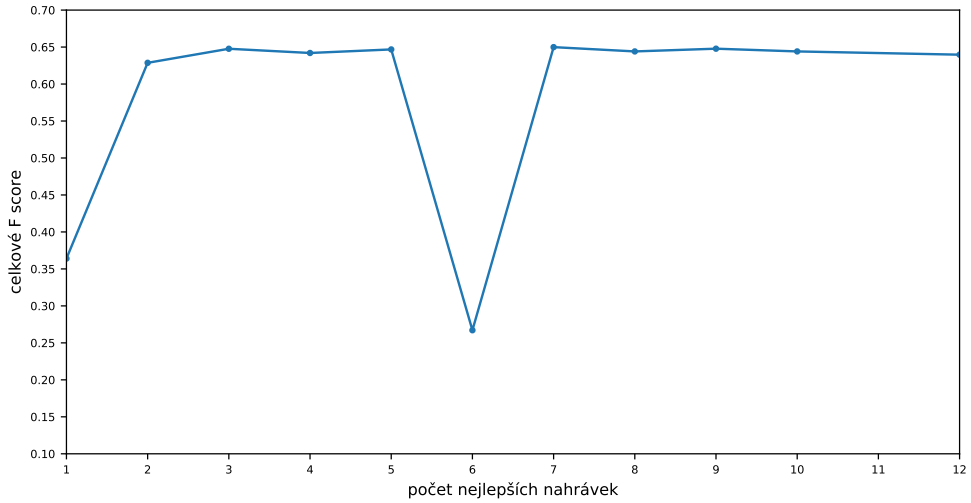
V provedených zarovnáních byl použit fonémový přepis `eng_advanced` pro angličtinu a `cze_advanced` pro češtinu. Hodnota `beam` byla nastavena na 300.

Nejprve se podívejme, jakých výsledků dosáhneme, pokud provedeme adaptaci celou testovací sadou (tj. 47 nahrávek). Důležité je rozhodnout o pořadí, ve kterém budeme nahrávky do modelu přidávat. Nabízí se pořadí určené podle hodnoty F score ze zarovnání s vlastními modely. Byly tedy vytvořeny dva modely ze všech nahrávek v sadě: od nejlepší po nejhorší (`47best model`) a naopak (`47worst model`). Tedy adaptace od nejlepší nahrávky po nejhorší a naopak. Výsledky vidíme na grafu 5.13. Model s 47 nejlepšími nahrávkami dosáhl podstatně horších výsledků než model s nahrávkami nejhoršími. Důvod je prostý, i když prvním případě začínáme s nejlepší nahrávkou, tak postupnou adaptací těmi horšími dochází k degradaci parametrů modelu. V druhém případě naopak sice začínáme s nekvalitním modelem, ale postupnou adaptací kvalitnějšími nahrávkami je model zlepšován.



Obrázek 5.13: Graf F score jednotlivých nahrávek s anglickým fonémovým přepisem zarovnaných modelem adaptovaným od nejlepší po nejhorší nahrávku (modrá) a od nejhorší po nejlepší (červená). Skórováno s přesností na 100 ms. Horizontální čáry reprezentují celkové F score.

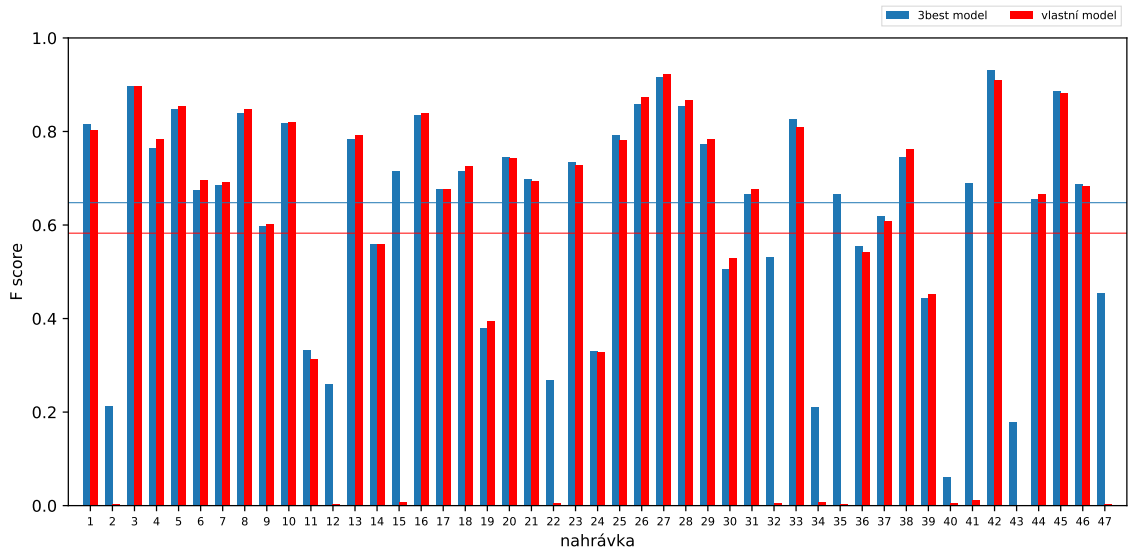
První pokusy s adaptací nebyly příliš úspěšné, dostali jsme ještě horší výsledky než v případě modelu z jediné nahrávky. Navíc použít všech 47 nahrávek trvá dlouhou dobu. Začneme tedy modelem z prvního experimentu této sady a budeme ho postupně adaptovat dalšími kvalitními nahrávkami v pořadí a uvidíme jak se celkové F score bude vyvíjet.



Obrázek 5.14: Vývoj celkového F score celé testovací sady v závislosti na počtu nejlepších nahrávek použitých pro adaptaci modelu, kterým byla sada zarovnána. Skórováno s přesností na 100 ms.

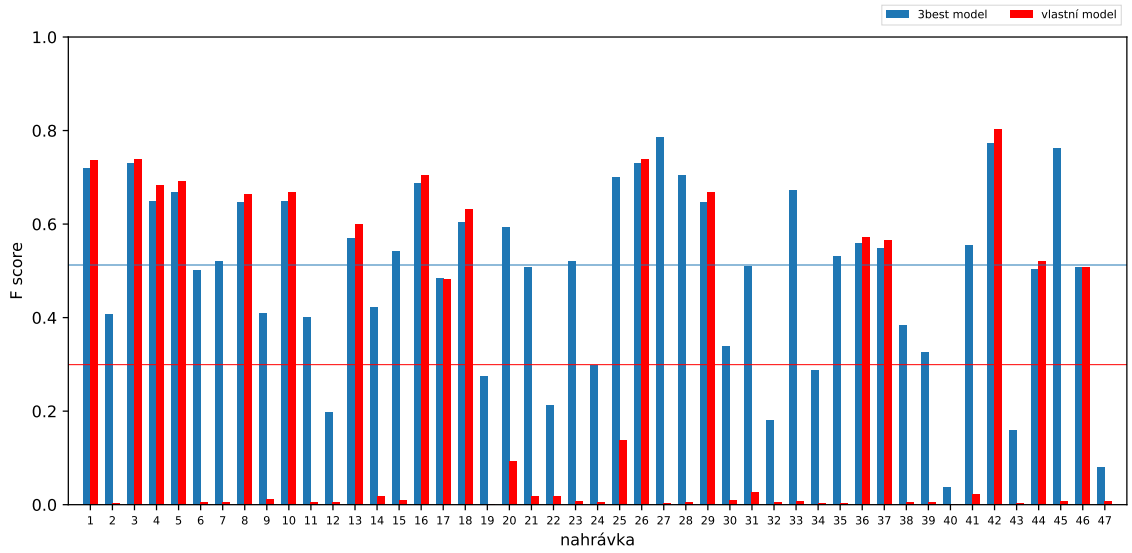
Na obrázku 5.14 se nachází vývoj celkového F score v závislosti na počtu nahrávek, seřazených od nejvyššího po nejnižší F score v zarovnání s vlastním modelem. Byly vytvořeny modely až po 12 nahrávek, kdy se zdá, že model začíná degradovat. Vidíme velké zlepšení mezi modelem jedné a dvou nahrávek, poté menší zlepšení se třemi nahrávkami. Dále se již kvalita zarovnání téměř vůbec nezlepšuje, dokonce vidíme že pro 6 nahrávek došlo k velké výchylce. Je možné že šestá nahrávka obsahovala nějaké nezvyklé kombinace fonémů a grafémů, které model zmátly. Modelem s 11 nahrávkami se dokonce nepodařilo celou sadu zarovnat z důvodu neznámého fonému. Lze očekávat, že se jedná o nějakou chybu zarovnávače, neboť s modelem z 12 nahrávek již problém nebyl.

I když nejvyššího celkového F score dosáhl model ze sedmi nahrávek, tak rozdíl oproti modelu ze tří je minimální. Ten navíc má také podstatně kratší časové nároky na vytvoření a zdá se být tedy optimální. Podívejme se na graf zarovnání s tímto modelem a srovnejme výsledky se zarovnáním z experimentu 5.1.3. Výsledný graf je na obrázku 5.15. Můžeme vidět, že zarovnání tímto modelem dosáhlo lepších celkových výsledků než zarovnání s vlastními modely. Výsledné celkové F score je po zaokrouhlení 0.65. U všech špatně oskórovaných nahrávek došlo k výraznému zlepšení a dokonce u některých dalších nahrávek jsme získali lepší skóre. U žádné nahrávky nedošlo k podstatnému zhoršení jako tomu bylo v experimentu 5.2.1, ale jen k drobným rozdílům.



Obrázek 5.15: Graf F score jednotlivých nahrávek s anglickým fonémovým přepisem zarovnaných modelem 3 nejlepších nahrávek (modrá) a vlastními modely nahrávek (červená). Skórováno s přesností na 100 ms. Horizontální čáry reprezentují celkové F score.

Na závěr tohoto experimentu ještě ověříme, jestli k takovým zlepšením dojde i v případě cizojazyčného, konkrétně českého, fonémového přepisu. Byl vytvořen český `3best model` podle výsledků experimentu 5.1.4. Výsledný graf je na obrázku 5.16. Došlo k výbornému zlepšení celkového F score na hodnotu dokonce větší jak 0.5. Chování výsledků jednotlivých nahrávek je stejné jako v případě anglických fonémových přepisů.



Obrázek 5.16: Graf F score jednotlivých nahrávek s českým fonémovým přepisem zarovnaných modelem 3 nejlepších nahrávek (modrá) a vlastními modely nahrávek (červená). Skórováno s přesností na 100 ms. Horizontální čáry reprezentují celkové F score.

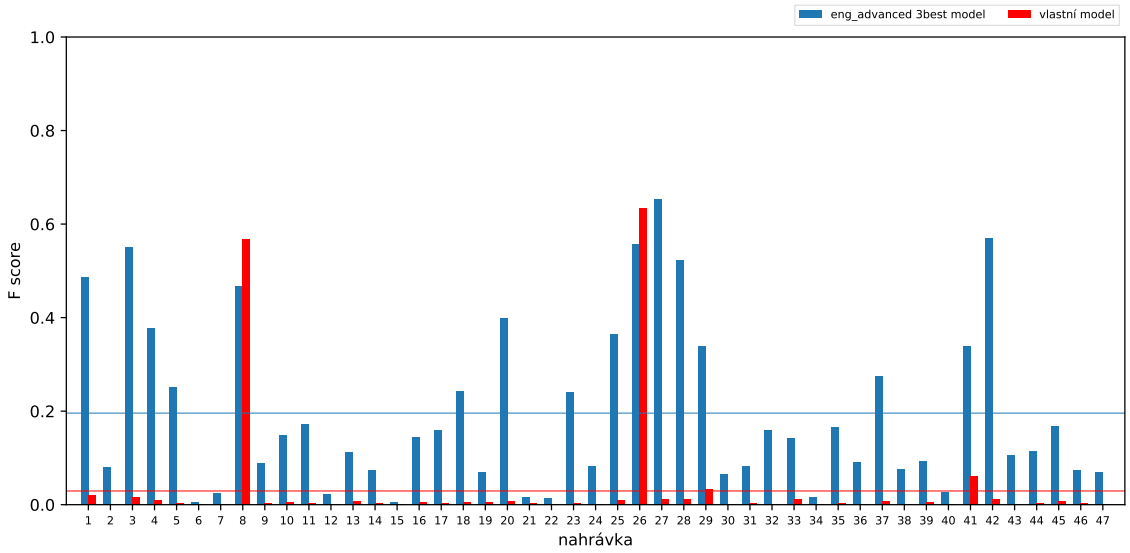
Pro maďarský fonémový přepis byl také vytvořen takový model, nebylo ovšem možné s ním zarovnat celou testovací sadu a to ani v případě, že bylo k vytvoření modelu použito všech 47 nahrávek. V každém případě docházelo ke ztrátě některých fonémů. Může se jednat o chybu nebo to může být vlivem automatického zmenšování počtu známých fonémů v modelu zarovnávačem.

5.2.3 Zarovnání nekvalitních fonémových přepisů modelem adaptovaným kvalitními přepisy

V tomto experimentu vezmeme `3best model` pro anglické fonémové přepisy `eng_advanced` a zarovnáme jím fonémové přepisy `eng`, u kterých jsme v experimentu 5.1.3 ukázali, že způsobují velmi špatné výsledky zarovnání.

Problémem, který bylo potřeba vyřešit, je rozdílná sada fonémů, které tyto dva systémy pro rozpoznávání používají. Běžné `eng` přepisy tedy nebylo možné tímto modelem zarovnat. Po důkladné kontrole slovníku fonémů byly odhaleny jen dva fonémy, které se liší. I když se nejednalo o pouhé jiné značení fonému, ale o skutečně rozdílné fonémy, byly tyto fonémy v `eng` přepisech zaměněny těmi z `eng_advanced`, aby bylo možnost provést alespoň nějaké zarovnání, které proběhlo standardně s hodnotou `beam` nastavenou na 300.

Výsledný graf je na obrázku 5.17. Pro srovnání jsou zobrazeny i výsledky původního zarovnání z experimentu 5.1.3. Vidíme, že došlo k zlepšení, ale bohužel stále celkové F score nedosahuje ani hodnoty 0.2. Toto zjištění jen potvrzuje velký vliv kvality fonémového rozpoznávání na výsledné zarovnání. Samotné chování jednotlivých nahrávek odpovídá tomu z předešlých experimentů s adaptovaným modelem - špatné nahrávky jsou výrazně zlepšeny a u dobrých nahrávek dochází k poklesu F score. V případě, že byly neshodné fonémy zaměněny v opačném pořadí, byly výsledky téměř identické.



Obrázek 5.17: Graf F score jednotlivých nahrávek s nekvalitním anglickým fonémovým přepisem `eng` zarovnaných modelem 3 nejlepších nahrávek kvalitního fonémového přepisu `eng_advanced` (modrá) a vlastními modely nahrávek (červená). Skórováno s přesností na 100 ms. Horizontální čáry reprezentují celkové F score.

5.3 Experimenty s modelem trénovaným na více nahrávkách

V poslední sadě experimentů se zaměříme na trénování modelu na více nahrávkách najednou a zarovnání s takto vytvořeným modelem. Tato funkcionalita není v `g2p_alignment.py` v původní verzi podporována a bylo nutné nástroj patřičně upravit. Tyto změny byly popsány v sekci 4.3.

V experimentech porovnáme výsledky trénování s výsledky s adaptací, zopakujeme neúspěšný experiment s maďarským fonémovým přepisem, který se pomocí adaptace nepodařilo provést a pokusíme se natrénovat vícejazyčný model.

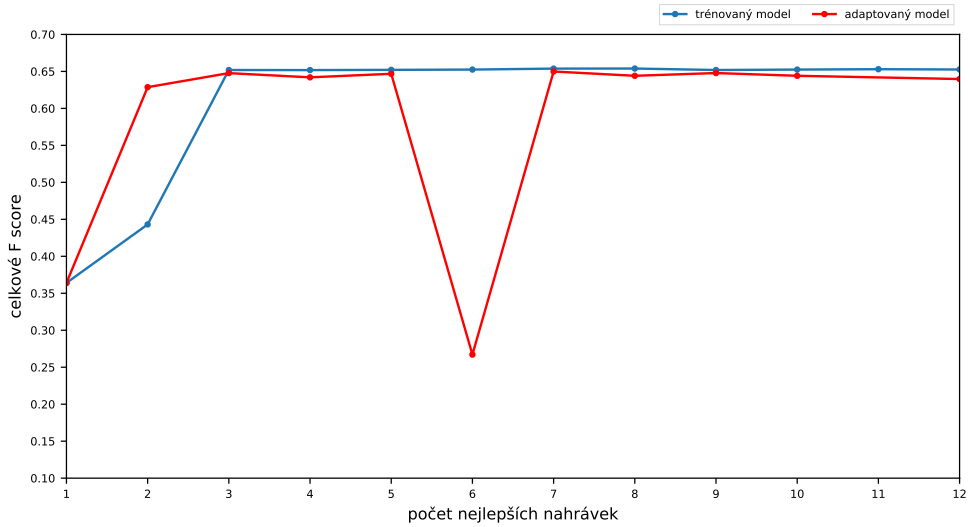
I když by pro samotné trénování (na rozdíl od adaptace) bylo možné použít i textový přepis `human_transcript`, není ovšem spolehlivě možné tímto modelem poté zarovnat další nahrávky z důvodu potencionálně neznámých grafémů. Proto pro všechny experimenty byl opět použit textový přepis `human_transcript_az`.

5.3.1 Zarovnání modelem trénovaným na n nahrávkách

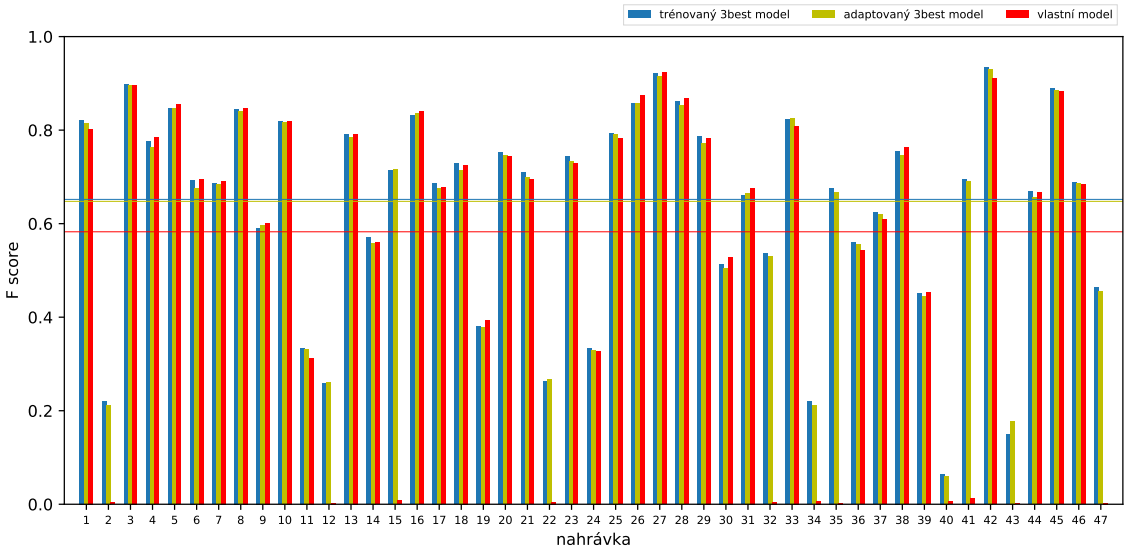
V prvním experimentu v této sadě zopakujeme experiment 5.2.1, jen místo adaptování modelu budeme trénovat modely n s nejlepšími nahrávkami najednou. Cílem je zjistit, jestli se trénování od adaptace liší nejen způsobem provedení, ale i co se výsledků zarovnání týká.

Vývoj celkové F score je na obrázku 5.18. Pro model z jedné nahrávky jsou výsledky samozřejmě stejné, v obou případech se jedná o zarovnání testovací sady modelem nejlepší nahrávky. Na rozdíl od adaptace, není zlepšení mezi jednou a dvěma nahrávkami tak drastické. Takového drastického zlepšení se dočkáme až při trénování na třech nahrávkách. Od tohoto počtu dále již celkové F score stagnuje. Můžeme vidět, že kromě prvních dvou modelů je celkové F score vždy slabě lepší, než tomu bylo v případě adaptace. Nedochází

také k nečekanému propadu F score v případě modelu trénovaném na šesti nahrávkách a všemi modely se podařilo úspěšně testovací sadu zarovnat.



Obrázek 5.18: Vývoj celkového F score pro celou testovací sadu v závislosti na počtu nejlepších nahrávek, které byly použity pro trénování (modrá) a adaptaci (červená) modelu. Skórováno s přesností na 100 ms.



Obrázek 5.19: Graf F score jednotlivých nahrávek s anglickým fonémovým přepisem `eng_advanced` zarovnaných modelem trénovaným na třech nejlepších nahrávkách (modrá), modelem adaptovaným na třech nejlepších nahrávkách (žlutá) a vlastními modely nahrávek (červená). Skórováno s přesností na 100 ms. Horizontální čáry reprezentují celkové F score.

Výsledky získané pomocí natrénovaných modelů na více nahrávkách se zdají být oproti adaptovaným modelům stabilnější. Stejně jako v případě adaptace již při třech nejlepších

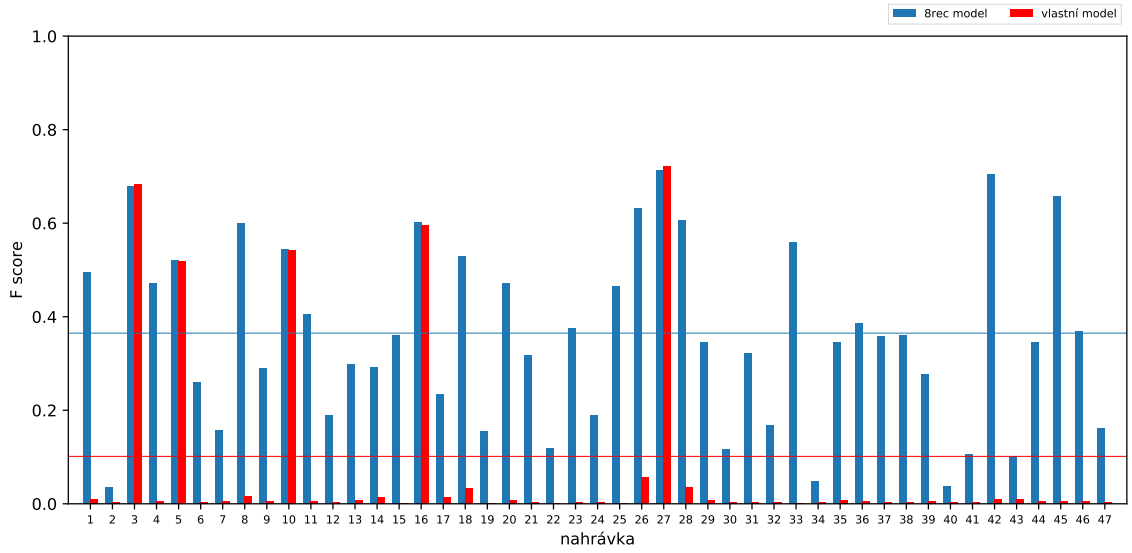
nahrávkách dostáváme optimální výsledek vzhledem k ostatním proměnným. Mezi tyto proměnné patří samozřejmě, stejně jako v případě adaptace, čas potřebný pro natrénování. Tento čas je ovšem o něco kratší, neboť je zarovnávač spuštěn pouze jednou. Hlavní překážkou v trénování modelů z více nahrávek je paměťová náročnost samotného trénování z důvodu vytváření velkých datových struktur během trénování. Klouzavou adaptací bylo možné vytvořit model 47 nejlepších nahrávek, trénováním by to vyžadovalo obrovské množství paměti RAM.

Na závěr tohoto experimentu se ještě podívejme na graf (obrázek 5.19) výsledků všech nahrávek pro model trénovaný na třech nejlepších nahrávkách v porovnání s odpovídajícím adaptovaným modelem a zarovnáním s vlastními modely. Je vidět, že F score není stabilně lepší pro všechny nahrávky a u některých dokonce dochází k mírnému poklesu.

5.3.2 Zarovnání maďarských fonémových přepisů

Při adaptaci se nepodařilo vytvořit model tří (a ani 47) nejlepších nahrávek s použitím maďarských fonémových přepisů z důvodu mizení některých fonémů z modelu. V případě trénování by k tomuto jevu docházet nemělo, ale stále nebylo možné natrénovat model ze tří nahrávek. Důvodem je prostý fakt, že první tři nahrávky neobsahují všechny fonémy které se v celé testovací sadě nacházejí. Z tohoto důvodu bylo ručně vybráno 8 co nejlepších nahrávek, které všechny potřebné fonémy obsahují. Vzniklý model označme `8rec model`.

Výsledek zarovnání je na grafu 5.20 spolu s původními výsledky zarovnání maďarských fonémových přepisů. Výsledky jsou podle očekávání o poznání lepší. Je důležité poznamenat, že výsledný model stále neobsahuje všechny fonémy, které použitý systém pro rozpoznávání zná a nemuselo by tedy být možné zarovnat tímto modelem libovolnou nahrávku.

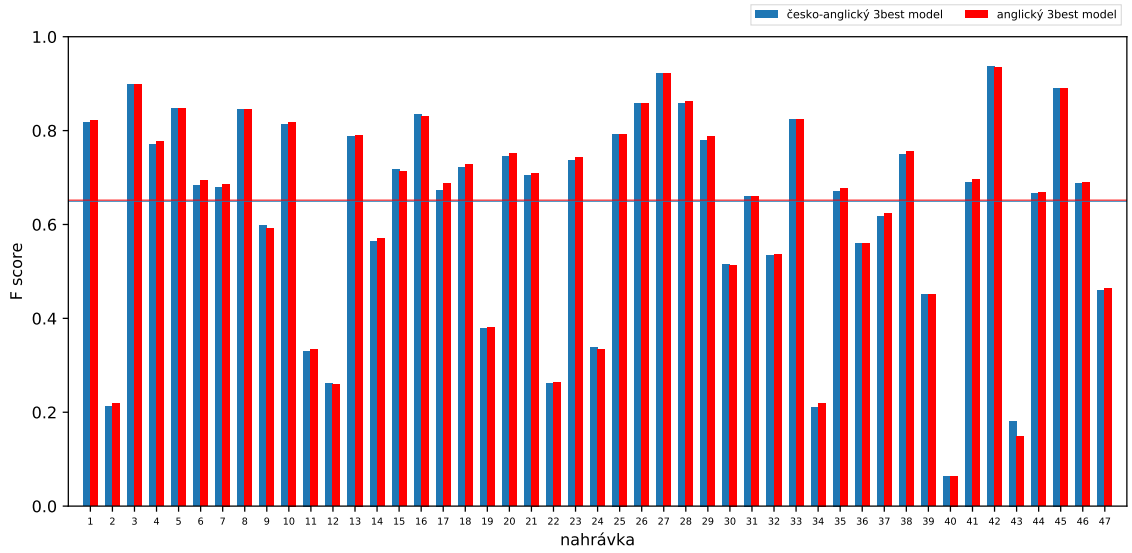


Obrázek 5.20: Graf F score jednotlivých nahrávek s maďarským fonémovým přepisem `hun` zarovnaných modelem z 8 nahrávek (modrá) a vlastními modely nahrávek (červená). Skórováno s přesností na 100 ms. Horizontální čáry reprezentují celkové F score.

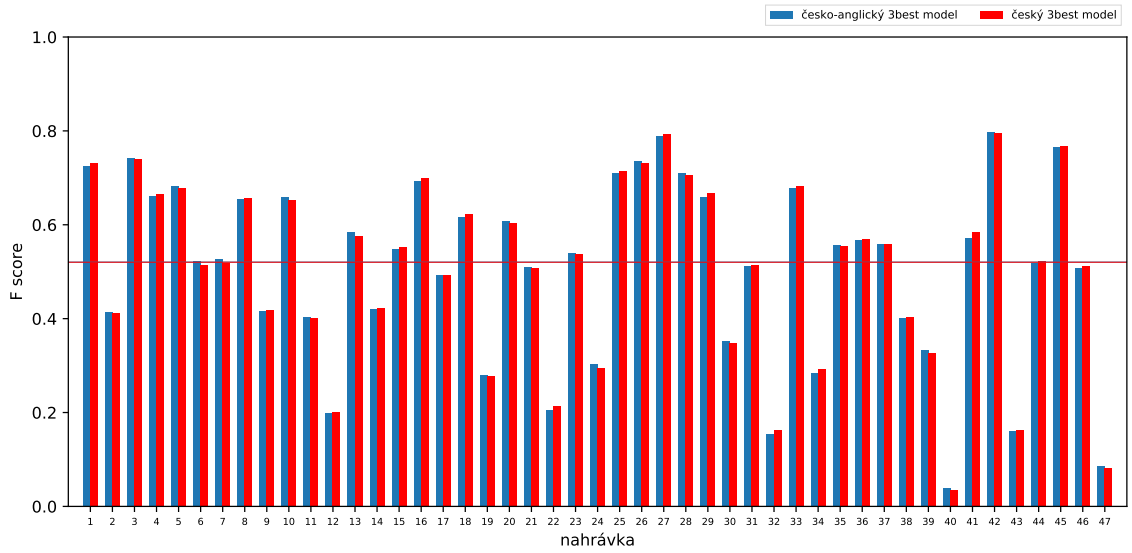
5.3.3 Vytvoření vícejazyčného modelu

V dřívějších experimentech jsme již ukázali, že i s použitím cizojazyčného systému pro rozpoznávání fonémů můžeme dosáhnout ne úplně špatných výsledků zarovnání. V tomto experimentu se pokusíme vytvořit model pro zarovnávání, který bude do jisté míry nezávislý na tom, jaký systém fonémového rozpoznávače použijeme.

Využijeme toho, že při trénování můžeme zarovnávači předat libovolné `.g2p.labels` soubory. Na vstup tedy předáme tři nejlepší nahrávky s anglickým fonémovým přepisem `eng_advanced` a tři nejlepší nahrávky s českým fonémovým přepisem `cze_advanced`. Výsledný model by měl poté znát jak české tak anglické fonémy. Mohli bychom dále přidávat tímto způsobem další jazyky a byli bychom omezeni jen pamětí potřebnou pro trénování na použitém množství nahrávek.



Obrázek 5.21: Graf F score jednotlivých nahrávek s anglickým fonémovým přepisem zarovnaných česko-anglickým `3best` modelem (modrá) a anglickým `3best` modelem (červená). Skórováno s přesností na 100 ms. Horizontální čáry reprezentují celkové F score.



Obrázek 5.22: Graf F score jednotlivých nahrávek s českým fonémovým přepisem zarovnaných česko-anglickým `3best` modelem (modrá) a českým `3best` modelem (červená). Skórováno s přesností na 100 ms.

Na obrázcích 5.21 a 5.22 vidíme grafy zarovnání anglických a českých fonémových přepisů vzniklým modelem. Jak v obou případě vidíme, výsledky jsou až na minimální rozdíly téměř identické k zarovnáním s běžnými modely trénovanými na třech nahrávkách, přičemž bylo použito pouze jediného modelu jak k anglickému tak českému zarovnání. Pokud budeme mít tedy tímto způsobem vytvořený model pro fonémy různých jazyků, můžeme jej poté použít pro libovolný fonémový přepis. Dále by mělo být možné vzít české nahrávky s českými textovými přepisy a použít pro trénování českých fonémů tyto `.g2p.labels` soubory. V takovém případě by se pak nahrávka mohla převést na fonémový přepis odpovídajícím systémem pro daný jazyk a poté by bylo možné použít jediný model pro zarovnání nahrávky v libovolném natrénovaném jazyce.

Obdobný experiment byl proveden i pro postupnou adaptaci modelu, zkoumanou v předchozí sadě experimentů. Bylo ovšem nutné anglickou a českou nahrávku spojit a vytvořit nové `.g2p.labels` soubory, aby byly všechny fonémy modelu známé již při jeho prvním vytvoření a následná adaptace byla možná. Tento způsob není vhodný z toho důvodu, že zarovnávač poté neví, kdy končí jedna nahrávka a začíná druhá v rámci jednoho `.g2p.labels` souboru. Mohlo by dojít k zarovnání fonémů jednoho jazyka na grafémy druhého a naopak a v důsledku toho by mohl být teoreticky vytvořen horší model.

5.4 Shrnutí výsledků

V této sekci shrneme provedené experimenty a podíváme se i na podstatné konkrétní hodnoty F score v tabulkách. Všechny zarovnání byly oskórovány s přesností 100 ms. V experimentech byla zjištěna optimální hodnota parametru `beam` zarovnávače, tak aby byl ušetřen čas a zároveň bylo dosaženo dostatečně kvalitních výsledků. Konkrétní F score pro různé hodnoty `beam` jsou v tabulce 5.1 spolu s průměrnými hodnotami F score ze tří krátkých a dlouhých nahrávek. Z těchto výsledků jsme zjistili, že i vzhledem k času potřebném pro za-

rovnání, je optimální hodnotou `beam` hodnota 300. F score se také mění rozdílně pro kratké a dlouhé nahrávky.

`beam`	celkové F score	čas (s)	krátké nahrávky[1]	dlouhé nahrávky[1]
10	0.0478	5474	0.0818	0.0213
50	0.2096	13682	0.2897	0.0810
100	0.3407	23065	0.3890	0.2174
200	0.4877	29500	0.5778	0.2653
300	0.5826	40942	0.5793	0.4677
400	0.6115	54546	0.5784	0.4920
500	0.6071	58931	0.5780	0.5043

[1] průměr F score tří nahrávek

Tabulka 5.1: Celkové F score a potřebný čas pro různé hodnoty `beam`.

Dále experimenty ukázaly, že kvalita použitého fonémového přepisu má obrovský vliv na kvalitu zarovnání. Příkladem může být srovnání kvalitních a nekvalitních anglických fonémových přepisů. Srovnání celkových F score lze nalézt v tabulce 5.2. Ve stejné tabulce je také srovnání různých grafémových přepisů pro kvalitních fonémový přepis `eng_advanced`.

fonémový přepis	grafémový přepis	celkové F score
`eng`	`human_transcript`	0.0293
`eng_advanced`	`human_transcript`	0.5826
	`human_transcript_az`	0.5138
	`reference`	0.5387

Tabulka 5.2: Celkové F score pro kvalitní a nekvalitní anglické fonémové a grafémové přepisy zarovnané s vlastními modely nahrávek.

V dalších experimentech bylo zkoumáno optimální množství nahrávek, které je vhodné použít pro vytvoření modelu z více nahrávek. Výsledky pro postupnou adaptaci modelu více nahrávkami a trénování modelu více nahrávkami najednou se nachází v tabulce 5.3. V případě adaptace docházelo k mnohem větším výchylkám v F score a pro jeden model se nepodařilo sadu vůbec zarovnat. Trénováním bylo dosaženo stabilnějších výsledků bez velkých výchylek. Na rozdíl od adaptace, která je jen časově náročná, má ovšem trénování nevýhodu v podobě vysoké paměťové náročnosti a je nevýhodné tímto způsobem vytvářet model z většího množství nahrávek. Pro oba způsoby je potřeba mít grafémové přepisy, které neobsahují zbytečné speciální symboly, obsažené jen v některých přepisech.

počet nahrávek	adaptovaný model	trénovaný model
1	0.3640	0.3640
2	0.6287	0.4431
3	0.6477	0.6519
4	0.6420	0.6518
5	0.6468	0.6521
6	0.2671	0.6525
7	0.6499	0.6538
8	0.6441	0.6539
9	0.6478	0.6519
10	0.6441	0.6525
11	-	0.6530
12	0.6397	0.6526

Tabulka 5.3: Celkové F score pro zarovnání s modely vytvořenými různým počtem nejlepších nahrávek použitých pro adaptaci a trénování modelu.

Bylo také experimentováno s jazykovou nezávislostí zarovnávače. Byly provedeny zarovnání s cizojazyčnými fonémovými přepisy, konkrétně s českými a maďarskými. Výsledky těchto experimentů jsou shrnuty v tabulce 5.4. Je vidět že podobně jako u anglických přepisů, adaptace a trénování dokáže zlepšit výsledky. Dalším jazykově nezávislým experimentech bylo vytvoření česko-anglického modelu. Výsledky v tabulce 5.5 ukazují, že zarovnání tímto modelem dosahují téměř identických výsledků jako zarovnání modelem vytvořeným pouze pro odpovídající jazyk. Teoreticky by tedy bylo možné vytvořit robustní model, který by byl schopen zarovnávat přepisy vytvořené různými fonémovými rozpoznávači. V praxi by tohle mělo význam, pokud bychom vytvořili takový model s použitím anglických a českých nahrávek, ne pouze s anglickými a českými přepisy anglických nahrávek.

fonémový přepis	vlastní modely	adaptovaný model	trénovaný model
`hun`	0.1014	-	0.3649[1]
`cze_advanced`	0.2994	0.5125[1]	0.5202[2]

[1] 3best model
[2] 8rec model

Tabulka 5.4: Celkové F score pro maďarské a české fonémové přepisy pro vlastní, adaptované a trénované modely.

fonémový přepis	jazykově odpovídající model	česko-anglický model
`eng_advanced`	0.6519	0.6495
`cze_advanced`	0.5202	0.5205

Tabulka 5.5: Celkové F score pro zarovnání s trénovanými jazykově odpovídajícími modely a česko-anglickým modelem.

Kapitola 6

Závěr

Práce popisuje omezení a překážky, se kterými je třeba počítat při návrhu nástroje pro zarovnávání audia a textu. Poskytuje krátký přehled vybraných prací zabývajících se touto problematikou. Dále poskytuje úvod do problematiky fonémového rozpoznávání a G2P konverze. Jsou shrnuty informace o MGB Challenge, ze které byla převzata data pro experimenty a způsob skórování, který je detailněji popsán.

Byl popsán postup práce se zkoumaným nástrojem v případě zarovnávání nahrávek, vytváření a postupná adaptace modelu pro zarovnávání více nahrávkami a trénování modelu více nahrávkami naráz. Byla podrobně popsána data použitá v experimentech.

Nakonec byly provedeny tři sady experimentů. První sada obsahovala experimenty se zarovnáním nahrávek jejich vlastními modely, tedy takovými, které byly trénovány jen na samotné jedné nahrávce. V těchto experimentech byla zjištěna optimální hodnota omezení šířky vyhledávání na hodnotu 300. Dále byl zkoumán vliv kvality fonémového přepisu na kvalitu zarovnání a vliv použití cizojazyčného systému pro rozpoznávání. Na závěr první sady experimentů bylo provedeno srovnání jednotlivých textových přepisů.

Druhá sada se zabývala postupnou adaptací modelu pro zarovnávání více nahrávkami. Byl nalezen optimální počet nahrávek vzhledem ke kvalitě výsledků a času potřebného pro zhotovení modelu. Dále bylo provedeno srovnání zarovnání provedených adaptovanými modely s výsledky z první sady experimentů. Poslední experiment ukázal zlepšení zarovnání nekvalitních fonémových přepisů v případě použití adaptovaného modelu z kvalitních přepisů.

Poslední sada experimentů zkoumala možnosti trénování modelu více nahrávkami zároveň. Bylo nutné provést úpravu dodaného nástroje tak, aby tato funkcionalita byla dostupná. Bylo provedeno srovnání kvality trénovaných modelů a těch adaptovaných z druhé sady experimentů. Za pomoci trénování byl vytvořen model maďarských fonémových přepisů, který nebylo možné vytvořit adaptací. Na závěr byl vytvořen česko-anglický model pro zarovnávání trénováním na anglických a českých fonémových přepisech naráz.

Experimenty také ukázaly, že nástroj má problém s tichými místy v nahrávkách, kdy fonémový rozpoznávač občas chybně rozezná některý foném ze šumu a v takovém případě zarovnávač dosahuje lepších výsledků, pokud je v textovém přepisu obsažen nějaký symbol, na který se může nadbytečné fonémy zarovnat. To stejné platí pro výskyt hudby a hluku v pozadí, kde se ukázalo, že titulky obsahující poznámky o hudbě nebo hluku, dosahují lepších výsledků než čistý přepis řeči.

Experimenty s cizojazyčnými přepisy ukázaly, že i když se adaptací a trénováním podařilo F score zlepšit, nejsou stále výsledky zdaleka dostačující a v praxi použití cizojazyčného

systému pro rozpoznávání není příliš užitečné. Větší potenciál má vytváření vícejazyčných modelů pro zarovnávání a jejich použití pro zarovnávání nahrávek v různých jazycích.

Další výzkum s tímto nástrojem by mohl obnášet úpravu textových přepisů do podoby zachovávající výhody původních přepisů, které obsahovaly speciální znaky reprezentující tichá místa a zároveň odstraňující nadbytečné speciální symboly, které zamezují adaptaci nebo trénování modelů s původními textovými přepisy. Dále by stálo za vyzkoušení experimentování s pořadím nahrávek při adaptaci, kde by mohlo být dosaženo lepších výsledků při adaptování v rámci n nahrávek od nejhorší po nejlepší. Při trénování v této práci nebyl kladen důraz na pořadí přidávání nahrávek jako příkladů pro trénování a změna pořadí by mohla také přinést lepší výsledky.

Význam má i zkoumání možností predikce kvality zarovnání z informací, které kromě samotného mapování fonémů na grafémy zarovnávač na výstupu poskytuje. V této oblasti by mohla najít uplatnění lineární regrese nebo využití neuronových sítí. Případně může být nutné získat ze zarovnávače další informace. Nezávisle na této práci vznikala bakalářská práce pana Šímy na stejné téma, který se ve své práci problematice predikce výsledků zarovnávače věnuje.

Ve vývoji je také nový nástroj pro G2P zarovnávání, který vychází z práce [7] a používá jiný způsob výpočtu pravděpodobností. Zde navržené a provedené experimenty mohou být poté zreplikovány s tímto novým nástrojem a výsledky porovnány pro ohodnocení kvality nového nástroje.

Literatura

[1] Ali, A.; Bell, P.; Glass, J.; aj. The MGB-2 challenge: Arabic multi-dialect broadcast media recognition. Dec 2016. 10.1109/SLT.2016.7846277.

[2] Ali, A. M.; Vogel, S.; Renals, S. : Speech Recognition Challenge in the Wild: Arabic MGB-3. *CoRR*, roč. abs/1709.07276, 2017. Dostupné z: <http://arxiv.org/abs/1709.07276>

[3] Anguera, X.; Perez, N.; Urruela, A.; aj. Automatic synchronization of electronic and audio books via TTS alignment and silence filtering. July 2011. ISSN 1945-7871, 10.1109/ICME.2011.6012185.

[4] Bell, P.; Gales, M. J. F.; Hain, T.; aj. The MGB challenge: Evaluating multi-genre broadcast media recognition. Dec 2015. 10.1109/ASRU.2015.7404863.

[5] Bisani, M.; Ney, H. : Joint-Sequence Models for Grapheme-to-Phoneme Conversion. roč. 50, 05 2008: s 434–451.

[6] Cvrček, V. *Precision a Recall* [online]. Příručka ČNK, Listopad 2014 [cit. 10.1.2018]. Dostupné z: <https://wiki.korpus.cz/doku.php/pojmy:precision>.

[7] Hannemann, M.; Trmal, J.; Ondel, L.; aj. Bayesian joint-sequence models for grapheme-to-phoneme conversion. March 2017. 10.1109/ICASSP.2017.7952674.

[8] Haubold, A.; Kender, J. R. Alignment of Speech to Highly Imperfect Text Transcriptions. July 2007. ISSN 1945-7871, 10.1109/ICME.2007.4284627.

[9] Hoffmann, S.; Pfister, B. : Text-to-speech alignment of long recordings using universal phone models. 01 2013: s 1520–1524.

[10] Manning, C. D.; Raghavan, P.; Hinrich, S. *Introduction to information retrieval.* Cambridge University Press, 2009. 58-60 s.

[11] Moreno, P. J.; Joerg, C.; van Thong, J. M.; aj. A Recursive Algorithm for the Forced Alignment of Very Long Audio Segments. 1998.

[12] Powers, D. : Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. roč. 2, 01 2008.

[13] Robert-Ribes, J.; Mukhtar, R. G. Automatic Generation of Hyperlinks Between Audio and Transcript. 1997.

[14] Schwarz, P. : *Phoneme recognition based on long temporal context.* Dizertační práce, 2009. Dostupné z: <http://www.fit.vutbr.cz/study/DP/PD.php?id=109>

Příloha A

Obsah přiloženého paměťového média

- `xsubaa00.pdf` - technická zpráva ve formátu PDF
- `tex` - adresář se zdrojovými soubory technické zprávy ve formátu LaTeX
- `alignment` - adresář s příkladem hotového zarovnání
- `results` - adresář s výsledky skórování provedených experimentů[1]
- `scripts` - adresář s použitými skripty
- `poster.pdf` - propagační plakát
- `2018-xsubaa00-synchronizace-textu-a-audia.mp4` - propagační video
- `video.xml` - popis videa ve formátu XML

[1] Z legálních důvodu nebylo možné přiložit hotová zarovnání z experimentů.